Learning to Rank Pre-trained Vision-Language Models for Downstream Tasks

Yuhe Ding, Bo Jiang, Aihua Zheng, Qin Xu and Jian Liang

Abstract—Vision language models (VLMs) like CLIP show stellar zero-shot capability on classification benchmarks. However, selecting the VLM with the highest performance on the unlabeled downstream task is non-trivial. Existing VLM selection methods focus on the class-name-only setting, relying on a supervised large-scale dataset and large language models, which may not be accessible or feasible during deployment. This paper introduces the problem of unsupervised vision-language model selection, where only unsupervised downstream datasets are available, with no additional information provided. To solve this problem, we propose a method termed Visual-tExtual Graph Alignment (VEGA), to select VLMs without any annotations by measuring the alignment of the VLM between the two modalities on the downstream task. VEGA is motivated by the pretraining paradigm of VLMs, which aligns features with the same semantics from the visual and textual modalities, thereby mapping both modalities into a shared representation space. Specifically, we first construct two graphs on the vision and textual features, respectively. VEGA is then defined as the overall similarity between the visual and textual graphs at both node and edge levels. Extensive experiments across three different benchmarks, covering a variety of application scenarios and downstream datasets, demonstrate that VEGA consistently provides reliable and accurate estimates of VLMs' performance on unlabeled downstream tasks.

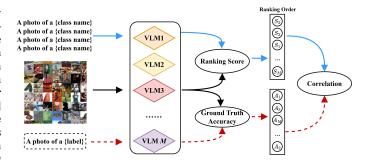*Index Terms*—Vision Language Model; Performance Evaluation

Fig. 1. Paradigm of unsupervised vision language model selection, where only unsupervised downstream datasets are available, with no additional information provided. The goal is to develop a method that computes a score for each VLM, which is highly correlated with the unseen ground truth accuracy.

I. INTRODUCTION

Vision language models (VLMs) like CLIP [1], ALIGN [2] and SigLIP [3], are transforming the technological and academic landscape with their unprecedented performance and the broad range of viable applications [4]–[7]. The most impressive capability of VLMs is their applications in zero-shot classification tasks. With just the class names, VLMs can be easily applied to any downstream task. However, identifying which VLM has the highest downstream performance is non-trivial, as labels are unavailable when deployed in real-world scenarios.

Recently, language-only vision language model selection (LOVM) [8], [9], which selects a VLM for the downstream dataset with only class names, has garnered attention. LOVM methods usually leverage the zero-shot classification accuracy on a large-scale dataset with annotations such as ImageNet [10] as a baseline, and additionally introduce large language

models (LLMs) [11] to generate captions and synonyms for these class names. However, the prediction results are sensitive to the quality of the content generated by LLM, and calling the LLM API can also be quite time-consuming and costly. Besides, a dataset with annotations is not always available during deployment. A viable solution is to use unsupervised downstream datasets along with the corresponding class names, which are readily accessible to downstream users in deployment scenarios. Some methods tailored for traditional convolution neural network models [12]–[14] also consider this problem. They typically predict downstream performance (also known as generalization performance or out-of-distribution performance) by measuring the distribution divergence between the training and downstream datasets. While this straightforward idea has been demonstrated to be applicable to VLMs [15], [16], implementing these methods directly on VLMs remains challenging. The reason is that training data is often difficult for downstream users to access, either due to its huge size or restrictions imposed by privacy and commercial considerations. Different from the two settings mentioned above, unsupervised vision language model selection aims to select VLMs using only the unlabeled target dataset. The paradigm is shown in Fig. 1, and this setting is practical and can eliminate the dependency on training datasets and LLMs presented in existing methods.

To solve this problem, we propose Visual-tExtual Graph Alignment (VEGA), a new method to evaluate the downstream performance of pre-trained VLMs with the corresponding unsupervised downstream dataset. VEGA is motivated by the pretraining paradigm of VLMs, which aligns features with the same semantics from the visual and textual modalities, thereby mapping both modalities into a shared representation space. In a well-trained cross-modality features space, visual features

Yuhe Ding, Bo Jiang, and Qin Xu are with the School of Computer Science and Technology, Anhui University. E-mail: madao3c@foxmail.com; jiangbo@ahu.edu.cn; xuqin@ahu.edu.cn.

Aihua Zheng is with the School of Artificial Intelligence, Anhui University. E-mail: ahzheng214@foxmail.com.

Jian Liang is with the New Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. E-mail: liangjian92@gmail.com.

Bo Jiang and Jian Liang are the corresponding authors.

should be tightly clustered around the corresponding textual features [1]. This phenomenon leads to a straightforward intuition: the more similar the structures of the class feature distributions for the two modalities, the easier it becomes to match the images to their corresponding class names. We model the structures of the class distributions in the two modalities as a fully connected visual graph and textual graph, respectively. Both graphs have the same number of nodes, with each node representing a class and edges representing the distances between connected classes. Specifically, the node and edge of the textual graph are simply defined as the textual feature of class names and the cosine distance, respectively. In the visual graph, nodes correspond to clusters of visual features of images, which are closest to the corresponding class name features, and edges represent the Bhattacharyya distances between the nodes. VEGA represents the similarity between the two graphs by combining both node and edge level similarity together. Specifically, node similarity is the average distance between image features in a visual node and the corresponding textual node. Edge similarity is the Pearson correlation coefficient between the edge matrices, which can eliminate the impact of scale. VLMs with a higher VEGA score are more likely to achieve better downstream performance.

We conduct extensive experiments on three practical application scenarios of VLM performance prediction, *i.e.*, VLMs from the CLIP family, VLMs from various pre-training algorithms, and the combination of VLM and prompt template. The results validate that VEGA is a reliable downstream performance indicator under various practical scenarios. The contributions of this study can be summarized as follows,

- We introduce a new problem setting that is practical for downstream users: unsupervised vision language model selection, where class names and unlabeled downstream datasets are available.
- We propose a novel method termed Visual-tExtual Graph Alignment (VEGA), which measures the similarity between the well-designed class distribution graphs of the visual and textual modalities, serving as an estimator of VLM zero-shot classification performance.
- We provide three benchmarks for this new setting, involving performance prediction on VLMs from the CLIP family, VLMs from various pre-training algorithms, and the combination of VLM and prompt template. Superior results validate that VEGA is a reliable unsupervised indicator of VLM downstream performance.

II. RELATED WORK

A. Model Selection

Model selection, a core challenge in transfer learning, focuses on ranking available pre-trained models to identify the one best suited for a given target task [17]–[19]. Model selection can be divided into several popular topics based on the different goals of the target task. Transferability estimation [18], [20], [21] aims to maximize the accuracy of the target task after supervised fine-tuning. The difficulty lies in how to select a model using a supervised target dataset

without the need for fine-tuning or a small amount of fine-tuning. Out-of-distribution (OOD) error prediction [17], [22] focuses on evaluating a model's ability to maintain robust performance when presented with data that deviates from its training distribution. These approaches involve using a test set specifically designed to include OOD data, allowing for an assessment of the model's generalization capacity under challenging, unseen conditions. Unlike traditional transfer learning approaches that require fine-tuning on downstream tasks, OOD error prediction remains within the same task framework, aiming to measure how well the model adapts to variations in data distributions without additional training. This evaluation provides insights into the model's resilience and reliability in real-world scenarios where data distribution shifts are inevitable. Model validation [23] is a crucial step in the machine learning workflow, enabling the evaluation and comparison of different training checkpoints to identify the most effective model. In supervised validation [23], a labeled validation set is used to measure performance and select the model with the best validation metrics, ensuring its ability to generalize to unseen data. In contrast, unsupervised validation [24]–[26] addresses scenarios where labeled validation data is unavailable. It leverages the unlabeled test set or proxy metrics to assess model performance, providing an alternative means for model selection in settings where labeling data is challenging or infeasible.

B. Vision-language Model Selection

LOVM [8] introduces a new setting termed language-only vision language model selection task, where methods are expected to perform both model selection and performance prediction based solely on a text description of the desired downstream application. LOVM generates a caption dataset and a synonym dataset and then calculates several statistic scores on these text datasets. This is an interesting setting and is reasonable in cases where data is extremely limited. However, LOVM relies on large language models (LLMs) [11] to generate a substantial number of captions and synonyms for these class names. The prediction results are sensitive to the quality of the content generated by the LLM, and calling the LLM API can also be quite time-consuming and costly. Besides, some recent studies [1], [15], [16] find that the generalization performance has a high correlation with train-test set similarity. They design various methods to measure train-test set similarity. For downstream users, the training set is difficult to obtain, while the downstream dataset is usually available. Therefore, developing a downstream performance evaluation method for vision language models with a downstream unsupervised dataset is practical.

C. Generalization Performance Prediction.

As the rapid proliferation of generalization algorithms such as domain generalization [17], distributionally robust optimization [27], invariant learning [28] and stable learning [29], etc, evaluating their ability under possible distribution shift scenarios becomes increasingly critical for a downstream application. Existing generalization performance prediction methods are

divided into several types. Confidence-based methods [12], [30] are based on the intuition that the performance of models is related to their prediction confidence. Discrepancy-based methods measure the distribution discrepancy between the training and test sets, with the aid of some classical metrics such as Frechec Distance [13] or well-designed methods such as projection norm [14]. Consistency-based methods measure the consistency of models under diverse scenarios and tasks [31], [32]. Actually, most generalization performance methods rely on the training data (also known as in-distribution, known distribution, source data, etc). However, for VLMs, the training data is huge, and some of it may be inaccessible due to privacy or commercial reasons, making it challenging to apply these methods directly to the performance prediction task of VLMs. We select four representative methods that do not strictly rely on training data and compare them in our experiments.

III. PRELIMINARY

In this section, we formally introduce the setting of unsupervised vision language model selection.

A. Zero-shot Classification of Vision Language Models

We denote the candidate VLMs as $\{v_m = (\phi_m, \xi_m)\}_{m=1}^M$, where ϕ_m and ξ_m notate the visual encoder and textual encoder of m-th VLM, respectively; $X = \{x_i\}_{i=1}^N$ denotes the unlabeled downstream dataset, where N is the number of images. $C = \{c_k\}_{k=1}^K$ represents the class names, i.e., label space, where K is the number of classes. Zero-shot classification with VLMs involves encoding both image and text prompts (e.g., "a photo of a {class name}") into feature vectors. An image is classified by selecting the class whose textual feature has the highest cosine similarity to the image's feature vector,

$$\hat{y}_i = \underset{k}{argmax}(cos(\xi_m(\tilde{c}_k), \phi_m(x_i))), \quad (1)$$

where $cos(\cdot)$ is the cosine similarity, $\tilde{c}_k$ is the text prompt of the class name c_k, y_i is the real label of x_i; $\phi_m(x_i) \in \mathbb{R}^D$ and $\xi_m(\tilde{c}_k) \in \mathbb{R}^{D \times K}$ denote the visual feature and textual feature respectively, D is the dimension of features. It is worth noting that, text prompts also play a crucial role when employing VLMs for zero-shot classification. Selecting an appropriate prompt template is essential, as it significantly impacts the effectiveness of zero-shot classification. Notate the templates as $\{\sigma_p\}_{p=1}^P$, P is the number of candidate templates, the text prompts $\tilde{c}_k$ are defined as $\tilde{c}_k = \sigma_i(c_k)$. For different VLMs, the optimal template is not necessarily the same, so it is equally important to choose a suitable combination of VLM and template.

B. Vision Language Model Selection

A large number of VLMs have emerged in recent years. There are dozens of different model architectures in the CLIP [1], [4] family alone, and diverse pre-training algorithms [3], [33] also flourished. Vision language model selection [8], [9] aims to select a model for downstream datasets with the highest zero-shot classification accuracy. Formally, a VLM

selection algorithm h aims to calculate a score s_m for each VLM $v_m = (\phi_m, \xi_m)$,

$$s_m = h(v_m) = h(\phi_m, \xi_m), \quad (2)$$

s_m is highly-correlated with the zero-shot performance $a_m = \frac{1}{N}\sum_{i=1}^N \mathbb{I}(\hat{y}_i = y_i)$, where $\hat{y}$ is the defined in Eq. (1), and y_i is the real label for each unlabeled image.

Language Only VLM Selection (LOVM). Existing methods focus on language-only VLM selection (LOVM) [8], where only meta information, i.e., class names, are available,

$$s_m = h_{LOVM}(v_m|C_d), \quad (3)$$

where C_d is the class name of dataset d. As information is scarce, LOVM introduces the large language model (LLM) [11], which is important prior knowledge to this setting. LLM generates many probable image captions, which could be encoded using the different VLM text encoders, producing the corresponding text embeddings, which are treated as image proxies. Existing work [8] introduces the accuracy of the candidate model on ImageNet [10] (INB) as a baseline, and additionally proposes the text classification score (LOVM-C) and dataset granularity score (LOVM-G). INB is a strong baseline, and its computation requires the full ImageNet dataset and its labels, which is not available in most real-world situations.

Unsupervised VLM Selection (UVMS). We focus on the Unsupervised VLM selection (UVMS) problem, where the unsupervised downstream data and the class names are available,

$$s_m = h_{UVMS}(\phi_m, \xi_m|C, X). \quad (4)$$

Due to the scarcity of supervision information, LOVM needs to introduce large-scale supervised datasets, i.e., ImageNet, and large language models. Our UVMS setting strictly requires only unsupervised downstream data to be available. This approach is more practical because we always have the test data during deployment, while the availability of a supervised dataset and LLMs is not guaranteed.

Evaluation of UVMS task. To evaluate the performance of the UVMS method comprehensively, we introduce four commonly used metrics in model evaluation methods [8], [24]. Specifically, given the ground truth, i.e., the zero-shot classification accuracy $\mathcal{A} = \{a_m\}_{m=1}^M$ of the candidate models on the target dataset, and the predicted scores $\mathcal{S} = \{s_m\}_{m=1}^M$ of the candidate models, the metrics are introduced as follows:

- **Top-5 Recall** (R_5). Top-5 recall measures the overlap between the five highest predicted models and the five actual optimal models,

$$R_5 = \frac{|F_5|}{5}, \quad F_5 = I(\mathcal{A}^5) \cap I(\mathcal{S}^5), \quad (5)$$

  where $I(\cdot)$ is the index set, $\mathcal{A}^5$ and $\mathcal{S}^5$ are top-5 values in $\mathcal{A}$ and $\mathcal{S}$, $\|F_5\|$ is the length of F_5.
- **Top-1 Accuracy.** Top-1 accuracy measures the reliability of a UVMS method when selecting a single model, which is defined as the ground truth accuracy of the model with the highest predicted score.
- **Kendall's Rank Correlation** (τ_5 and τ). We use Kendall's rank correlation to evaluate the overall ranking
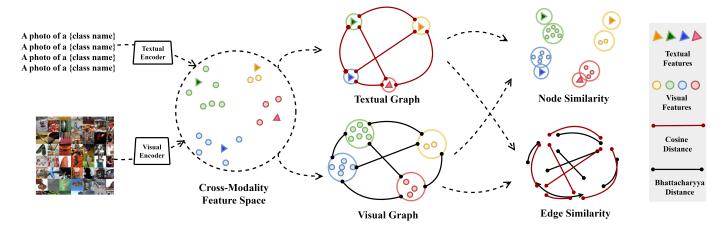
Fig. 2. The pipeline of VEGA involves encoding class names and unlabeled images into a shared cross-modality feature space. Subsequently, we construct a textual graph and a visual graph for the two modalities, respectively. VEGA combines node-level and edge-level similarities to evaluate the alignment between these graphs.

ability of the UVMS method on the best five models and the entire model zoo,

$$\tau = \frac{2}{|F|(|F|-1)} \sum_{i<j, i,j\in F} \text{sign}(a_i - a_j)\,\text{sign}(s_i - s_j),$$

$$\tau_5 = \begin{cases} 0, & \text{if } |F_5| < 2, \\ \frac{2}{|F_5|(|F_5|-1)} \sum_{\substack{i<j \\ i,j\in F_5}} \text{sign}(a_i - a_j)\,\text{sign}(s_i - s_j), \\ \quad \text{otherwise.} \end{cases}$$

(6)

where $F = \{1, 2, ..., M\}$, $F_5 = I(\mathcal{A}^5) \cap I(\mathcal{S}^5)$, and $\text{sign}(\cdot)$ is the sign function.

IV. METHOD

A. Motivation

Vision language models have flourished in recent years [1], [3], [33]. The classical VLM pre-pretraining paradigm is based on contrastive learning techniques. NT-Xent loss is extended to the multimodal domain,

$$\mathcal{L}_{\text{VLM}} = -\frac{1}{2}\,\mathbb{E}_{(x,y)\sim P_{\text{data}}, \{x_i', y_i'\}_{i=1}^N \sim P_{\text{data}}} \left[\log \frac{\exp(x^\top y/\tau)}{\sum_i \exp(x_i'^\top y/\tau)} + \log \frac{\exp(x^\top y/\tau)}{\sum_i \exp(x^\top y_i'/\tau)}\right].$$

(7)

$\mathcal{L}_{VLM}$ aligns the positive pair of text y and the corresponding image x. Concurrently, N negative pairs are denoted as $\{x_i', y_i'\}$. Both positive and negative pairs are sampled from the original data distribution P_{data}. With contrastive pre-training, the features of both modalities are mapped into a shared representation space, where images and texts with the same semantics are clustered together. Zero-shot classification is all about selecting the most recent class name for an image. In the process of modal alignment, the performance of zero-shot classification is gradually improved. Therefore, the performance of the VLM can be estimated by measuring the modality gap, *i.e.*, the alignment level between modalities.

B. VEGA: Visual-Textual Graph Alignment for Unsupervised VLM Selection

The key idea behind our method is that in the shared cross-modality feature space, the more similar the structures of the class feature distributions are between the two modalities, the easier it becomes to match images with their corresponding classes. Based on this intuition, we propose Visual-tExtual Graph Alignment (VEGA) to measure the similarity between these structures. The pipeline of VEGA is shown in Fig. 1. The class names are transformed into text prompts, which, along with unlabeled images, are encoded using the textual and visual encoders, respectively. We then represent the structure of the class feature distributions for the two modalities as a textual graph and a vision graph. VEGA is defined as the similarity between these two graphs. The key challenge of VEGA is constructing modality-specific class distribution graphs and measuring their similarity. We will elaborate on these details in the following sections.

Textual Graph. Given the limited information available from the textual modality, we represent the nodes directly as the text features of each class and the edges as the cosine similarity between each pair of nodes. Formally, the fully connected textual graph is denoted by $G_T = \{N_T, E_T\}$, where $N_T = \{n_k^T\}_{k=1}^K$ represents the nodes, and $E_T = \{e_{ij}^T\}_{i,j=1}^K$ represents the edges. Specifically, $n_k^T = \xi(\tilde{c}_k)$ denotes the node features, and $e_{ij}^T = cos(\xi(\tilde{c}_i), \xi(\tilde{c}_j))$ denotes the edge weights, calculated as the cosine similarity between the textual features.

Visual Graph. Modeling the visual graph $G_V = \{N_V, E_V\}$, is more complex than modeling the the textual graph. Nodes cannot be represented by a single vector for two reasons. First, a single vector lacks the capacity to fully represent a class. Second, without labels, it is challenging to determine which class an image belongs to. Therefore, in the cross-modal feature space, we use K textual features as centers to partition the visual features into K clusters. The concatenation of features within each cluster represents a node:

$$n_k^V = cat(\{\phi(x_i) \cdot \mathbb{I}(\hat{y}_i = c_k)\}_{i=1}^N),$$

(8)

where $\hat{y}_i = \underset{k}{argmax}(cos(\xi_m(\tilde{c}_k), \phi_m(x_i)))$, and $cat(\cdot)$ denotes concatenation. Since the number of visual features in each class cluster varies, node sizes differ, making it unsuitable to use a simple cosine distance for calculating edges. To address this issue, we model each class as a Gaussian distribution $\mathcal{N}_k$ with class means $\overline{n}_k^V$ and covariance Σ_k. $\overline{n}_k^V$ is the mean vector of n_k^V, and Σ_k is the covariance matrix,

$$\overline{n}_k^V = \frac{1}{N_k} \sum_{i=1}^{N} \phi(x_i) \cdot \mathbb{I}(\hat{y}_i = c_k),$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N} \left(\phi(x_i) - \overline{n}_k^V \right) \left(\phi(x_i) - \overline{n}_k^V \right)^\top \cdot \mathbb{I}(\hat{y}_i = c_k),$$

(9)

where $N_k = \sum_{i=1}^{N} \mathbb{I}(\hat{y}_i = c_k)$ is the number of features in the cluster of c_k. Each edge e_{ij}^V in $E_V = \{e_{ij}^V\}_{i,j=1}^K$ is defined by the Bhattacharyya coefficient between each pair of class Gaussians,

$$e_{ij}^V = Bh(\mathcal{N}_i, \mathcal{N}_j)$$
$$= \frac{1}{8} \left(\overline{n}_i^V - \overline{n}_j^V \right)^\top \Sigma^{-1} \left(\overline{n}_i^V - \overline{n}_j^V \right) + \frac{1}{2} \ln \frac{|\Sigma|}{\sqrt{|\Sigma_i| |\Sigma_j|}},$$

(10)

where $\Sigma = \frac{1}{2}(\Sigma_i + \Sigma_j)$, $|\cdot|$ denotes determinant. Using distributional distance as the edge measure, rather than the distance between single vectors like class means, more accurately represents the relationships between classes. This approach accounts for within-class covariance, capturing the dispersion of features within each class.

Cross-Modality Graph Similarity. Finally, the VEGA score s is defined as the summation of the node similarity s_n and edge similarity s_e. Node similarity is determined by the weighted average distance from all visual features within a cluster to the corresponding textual feature,

$$s_n = \frac{1}{K} \sum_{i=1}^{K} sim(n_k^T, n_k^V) \cdot N_k,$$

(11)

where $N_k = \sum_{i=1}^{N} \mathbb{I}(\hat{y}_i = c_k)$. Considering the different scales of various VLM features, we normalize each feature at the class level to obtain relative distances,

$$sim(n_k^T, n_k^V) =$$
$$\frac{1}{N_k} \sum_{i=1}^{N} \frac{exp\left(cos(\phi(x_i), \xi(c_k))/t\right)}{\sum_{k'=1}^{K} exp\left(cos(\phi(x_i), \xi(c_{k'}))/t\right)} \cdot \mathbb{I}(\hat{y}_i = c_k),$$

(12)

where $exp(\cdot)$ is the exponential function, and $t = 0.05$ is a temperature parameter in the normalization. For any VLM, the range of node similarity s_n is constrained to the range of 0 to 1. Similarly, due to the scale differences between the Bhattacharyya coefficient and cosine distances, we use the Pearson correlation coefficient [34] to measure edge similarity,

$$corr(E_T, E_V) = \frac{\sum_{i=1}^{K^2}(e_i^V - \overline{e}^V)(e_i^T - \overline{e}^T)}{\sqrt{\sum_{i=1}^{K^2}(e_i^V - \overline{e}^V)^2}\sqrt{\sum_{i=1}^{K^2}(e_i^T - \overline{e}^T)^2}},$$

(13)

where e_i is i-th element in E, $\overline{e}^V$ and $\overline{e}^T$ denote the mean value of E_V and E_T, respectively. Since the Pearson correlation coefficient ranges from -1 to 1, we re-scale s_e to a range of 0 to 1 to avoid the trade-off between s_n and s_e,

$$s_e = \frac{1}{2} \cdot corr(E_T, E_V) + \frac{1}{2}.$$

(14)

The formulation of VEGA is a simple summation of the two similarities: $s = s_n + s_e$. VEGA is a user-friendly method, as its implementation requires no backward propagation process and does not rely on LLMs. It involves only a small amount of inference and computation, making it easy to implement on general mid-range to low-end GPUs and CPUs.

V. EXPERIMENTS

We construct three benchmarks across three practical application scenarios for VLM performance evaluation, including performance prediction for VLMs from the CLIP family and various other pre-training algorithms respectively; and ranking the combinations of VLM and prompt templates.

Downstream Datasets. We conduct performance prediction on ten common-used downstream datasets, including basic image recognition Cifar-100 [35]; animal and plant dataset Oxford Pets [36] and Oxford Flowers [37]; street scene dataset SVHN [38] and GTSRB [39]; describable textures dataset DTD [40]; scene classification dataset Country211 [1], [41] and SUN397 [42]; digit dataset MNIST [43]; and facial expression dataset Fer2013 [44].

Baselines. We compare our method with existing training data-free methods in the fields of generalization error prediction [30], [31], [45], unsupervised model validation [24], and vision language model selection [8]. These methods are highly related to our setting, which could evaluate the performance without training data and the annotations of downstream datasets.

• **Entropy (ENT)** is a commonly used baseline, representing the entropy of the probability distribution of the logits from VLMs,

$$s_{ENT} = -\frac{1}{N} \sum_{i=1}^{N} P(x_i) \log P(x_i),$$
$$\text{where} \quad P(x_i) = \frac{exp\left(cos(\phi(x_i), \xi(c_k))\right)}{\sum_{k'=1}^{K} exp\left(cos(\phi(x_i), \xi(c_{k'}))\right)}.$$

(15)

• **Confidence Score (Conf)** [30] is a classical confidence-based method, defined as the average highest confidence score,

$$s_{Conf} = \frac{1}{N} \sum_{i=1}^{N} max(\{P(x_i)[k]\}_{k=1}^K).$$

(16)

• **Rotation (Rot)** [31] is inspired by self-supervised methods [46] and uses the accuracy of rotation angle prediction as a metric,

$$s_{Rot} = \frac{1}{4N} \sum_{i=1}^{4N} \mathbb{I}(\hat{y}_i^r = y_i^r),$$
$$\hat{y}_i^r = \underset{k}{argmax}(cos(\xi(Y_k^r), \phi(x_i))),$$

(17)

where the label space Y^r is defined as $\{0, 90, 180, 270\}$, and the template is " An image rotated by y_i^r degrees". Each image

is augmented to obtain four rotated images. Note that Rot can be used to select different image encoders, but not prompt templates, as image rotation does not involve text encoders.

• **SND** [24] is designed for unsupervised validation and is defined as neighborhood density,

$$s_{SND} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N}D_{ij}\log D_{ij},$$
$$D_{ij} = \frac{exp(N_{ij}/\tau)}{\sum_{j'}exp(N_{ij'}/\tau)}, \qquad (18)$$

where N_{ij} is the cosine distance between i-th image and j-th image, *i.e.*, soft neighbor distance. SND measures the soft neighbor density of a representation space.

• **Dispersion Score (DS)** [45] performs unsupervised clustering on the target dataset, and measures the separability among class means,

$$s_{DS} = \log\frac{\sum_{k=1}^{K}n_k\cdot\|\bar{\boldsymbol{\mu}}-\widetilde{\boldsymbol{\mu}}_k\|_2^2}{K-1}, \qquad (19)$$

where n_k is the number of k-th cluster, μ_k is the k-th cluster center, and $\bar{\mu}$ is the center of cluster center.

• **LOVM-G** and **LOVM-C** [8] are the dataset granularity score and the text classification score respectively, which measure the dataset difficulty and class clarity. LOVM-C leverages the generated captions dataset as image proxies and replace the images with the generated image captions to calculate each VLM's text top-1 accuracy and f1-score. LOVM-G includes three metrics: The Fisher criterion [47] evaluates the similarity and separation between classes; the Silhouette score [48] quantifies the compactness of same-class samples relative to the separation of different-class samples; and Class Dispersion score, which is their normalization constant, measures the tightness within a single class or the radius of its data cone. More details can be found in [8].

Details. All the experiments are conducted on NVIDIA Geforce 3090Ti GPU, and the temperature in Eq. (12) set to 0.05 across all the experiments. There are no random operations involved in our experiments. The Python implementation of VEGA and the specific predicted scores for all quantitative results are provided in the supplementary material.

A. Performance Prediction for VLMs from CLIP Family

CLIP [1] is the most popular VLM in recent years, with many CLIP models trained on various architectures and source datasets available as open-source. We first examine the evaluation of the CLIP family across different network architectures and source datasets.

Candidate Models. We have collected 31 CLIP models with diverse architectures and source datasets from OpenCLIP [1]. These models are the same as those used in LOVM [8] and include architectures from model families such as ResNet [49], ViT [50], and ConvNeXt [51], with source datasets comprising various versions of LAION [52]. Detailed information on the candidate models is provided in the supplementary material. For all candidate models, we use several commonly used

prompt templates [8] and compute the mean to obtain the textual feature.

Quantitative Results. We provide the complete results in Table I, where red indicates the best result in each row, and green represents the second-best. The table showcases the effectiveness of different methods in predicting downstream performance across various datasets on the CLIP family. Specifically, VEGA achieves the highest average scores for both Top-5 recall (R_5) and overall Kendall correlation (τ), with values of 0.64 and 0.62, respectively, showcasing its robustness in model selection. Notably, VEGA consistently ranks first or second in most datasets, including Flowers, GTSRB, and OxfordPets, where accurate predictions are critical for selecting the best-performing models. Additionally, VEGA's Top-1 accuracy aligns closely with the Oracle results, further validating its reliability in identifying models with superior downstream performance. These results highlight VEGA as a state-of-the-art, user-friendly approach for VLM selection and performance prediction. SND shows a negative correlation with downstream tasks, which might be because SND is designed for unsupervised validation tasks. However, in VLM model selection, where the differences between models are often more pronounced, a higher SND might indicate that the model has not learned a clear decision boundary.

Qualitative Results. We visualize the correlation between the actual downstream accuracy and the predicted scores for various methods in Fig. 3. The overall trend is basically consistent with that of quantitative experiments. Our method shows the strongest correlation, with data points closely following a linear trend, indicating high predictive accuracy. In contrast, methods like Entropy, Confidence Score, and Dispersion Score exhibit moderate correlations with more scattered data points, reflecting less consistent predictive power. Rotation and SND display the weakest correlations, with widely dispersed points and no clear linear pattern.

B. Performance Prediction for VLMs from Various Pre-training Algorithms

Recent advancements in VLM algorithms have created a range of options for users. When selecting an algorithm for zero-shot classification, which typically involves a standard network structure (visual and textual encoders), performance prediction methods can be applied similarly to those used for CLIP. LOVM-G and LOVM-C are not compared because they did not provide the caption and synonym datasets generated by LLMs. In our experiments, we found that the effects of LOVM-G and LOVM-C are sensitive to the caption and synonym datasets, so we cannot guarantee the reliability of our reproducible results.

Candidate Models. We collect 17 models from Hugging Face [2] from 10 commonly used VLM pre-training algorithms on zero-shot classification, including ALIGN [2], AltCLIP [53], CLIP [1], GroupViT [54], SigLIP [3], StreetCLIP [55], Meta-CLIP [56], BiomedCLIP [57], QuiltNet [58], BioCLIP [59]. For each algorithm, we select two models, except for those methods that only have one official open-source model. In

---

[1] https://github.com/mlfoundations/open_clip

[2] https://huggingface.co

TABLE I
DOWNSTREAM ZERO-SHOT CLASSIFICATION PERFORMANCE PREDICTION ON CLIP MODELS WITH VARIOUS ARCHITECTURES AND SOURCE DATASETS. RED INDICATES THE BEST RESULT IN EACH ROW, AND GREEN REPRESENTS THE SECOND-BEST. ORACLE IS THE BEST ACCURACY IN THE CANDIDATE MODEL, WHICH IS THE UPPER BOUND OF TOP-1 ACCURACY.

Dataset	ENT	Conf	Rot	SND	DS	LOVM-G	LOVM-C	VEGA	ENT	Conf	Rot	SND	DS	LOVM-G	LOVM-C	VEGA
					R_5								τ_5			
Cifar100	0.60	0.60	0.00	0.00	0.40	0.40	0.20	0.80	-1.00	-0.33	0.00	0.00	1.00	1.00	0.00	1.00
Country211	0.40	0.60	0.40	0.00	0.00	0.20	0.20	0.60	-1.00	-0.33	1.00	0.00	0.00	0.00	0.00	0.33
DTD	0.60	0.80	0.20	0.00	0.00	0.20	0.60	0.80	-1.00	-0.33	0.00	0.00	0.00	0.00	1.00	0.67
Flowers	0.60	0.60	0.40	0.00	0.00	0.00	0.00	0.80	-0.33	0.33	-1.00	0.00	0.00	0.00	0.00	0.67
GTSRB	0.60	0.60	0.20	0.00	0.20	0.60	0.60	0.60	-0.33	0.33	0.00	0.00	0.00	-0.33	0.33	-0.33
MNIST	0.00	0.20	0.40	0.00	0.20	0.00	0.20	0.40	0.00	-0.33	-1.00	0.00	0.00	0.00	0.00	1.00
Pets	0.60	0.60	0.20	0.00	0.00	0.00	0.00	0.80	0.33	-0.33	0.00	0.00	0.00	0.00	0.00	0.67
SVHN	0.20	0.40	0.20	0.00	0.20	0.20	0.00	0.80	0.00	-1.00	0.00	0.00	0.00	0.00	0.00	0.00
SUN397	0.40	0.60	0.00	0.00	0.00	0.00	0.40	0.40	1.00	1.00	0.00	0.00	0.00	0.00	1.00	-1.00
Fer2013	0.60	0.60	0.20	0.00	0.60	0.60	0.20	0.40	-0.33	-0.33	0.00	0.00	-1.00	-1.00	0.00	-1.00
Avg.	0.46	0.56	0.22	0.00	0.16	0.22	0.24	0.64	-0.27	-0.10	-0.10	0.00	0.00	-0.03	0.23	0.20

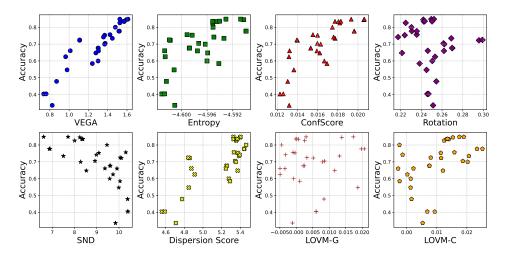| Dataset	ENT	Conf	Rot	SND	DS	LOVM-G	LOVM-C	VEGA	ENT	Conf	Rot	SND	DS	LOVM-G	LOVM-C	VEGA	Oracle
					τ								Top-1 Acc.				
Cifar100	0.51	0.63	-0.09	-0.54	0.54	0.09	0.34	0.81	0.78	0.85	0.72	0.40	0.80	0.78	0.78	0.85	0.85
Country211	0.48	0.59	0.17	-0.32	0.16	0.19	0.51	0.57	0.29	0.30	0.21	0.15	0.22	0.22	0.30	0.30	0.33
DTD	0.57	0.69	0.11	-0.43	0.45	0.15	0.53	0.77	0.66	0.66	0.53	0.35	0.59	0.58	0.68	0.68	0.68
Flowers	0.50	0.62	0.15	-0.26	0.32	0.06	0.07	0.66	0.80	0.80	0.73	0.55	0.72	0.73	0.58	0.81	0.81
GTSRB	0.36	0.48	0.09	-0.17	0.34	0.44	0.47	0.46	0.47	0.50	0.47	0.29	0.49	0.44	0.47	0.50	0.55
MNIST	0.26	0.34	0.07	-0.11	0.20	0.22	0.46	0.50	0.56	0.56	0.29	0.66	0.69	0.69	0.73	0.58	0.76
Pets	0.56	0.64	0.25	-0.44	0.37	0.04	0.05	0.74	0.92	0.92	0.91	0.75	0.88	0.88	0.91	0.93	0.93
SVHN	0.47	0.44	-0.19	-0.33	0.38	0.38	-0.05	0.56	0.46	0.46	0.34	0.16	0.46	0.50	0.45	0.46	0.56
SUN397	0.62	0.72	-0.34	-0.49	0.30	0.10	0.41	0.78	0.76	0.76	0.64	0.50	0.69	0.72	0.72	0.76	0.76
Fer2013	0.32	0.37	0.10	-0.35	0.28	0.04	0.44	0.33	0.28	0.33	0.32	0.23	0.28	0.33	0.32	0.34	0.34
Avg.	0.46	0.55	0.03	-0.35	0.33	0.17	0.32	0.62	0.60	0.62	0.52	0.40	0.58	0.59	0.59	0.62	0.66

Fig. 3. Visualization of the correlation between the actual zero-shot classification accuracy and predicted scores for various VLMs in the CLIP family.

total, there are 17 models, with specific information provided in the supplementary material.

Quantitative Results. We compare the performance of various methods in predicting downstream performance based on the pre-training algorithms of VLMs in Table II. VEGA achieves the highest average performance across four metrics. The baseline method Confidence Score also performs well on several simple datasets, likely because the dataset has smaller inter-class differences, leading to higher model uncertainty. In contrast, other methods exhibit weaker and less consistent performance. A high R_5 score (0.52) for Rotation, combined

with average performance on other metrics, indicates that rotation is relatively reliable when selecting a few high-performing models. SND still shows a negative average correlation (τ of -0.30), indicating poor alignment with actual downstream results. DS and ENT perform better than SND, but they are not as effective as Rot and VEGA. Overall, VEGA's strong and consistent performance across various datasets underscores its effectiveness in predicting the downstream impact of VLM pre-training algorithms, making it a more reliable and accurate method compared to its counterparts.

Qualitative Results. The visualization results are shown in

TABLE II
DOWNSTREAM ZERO-SHOT CLASSIFICATION PERFORMANCE PREDICTION FOR VLMs FROM VARIOUS PRE-TRAINING ALGORITHMS.

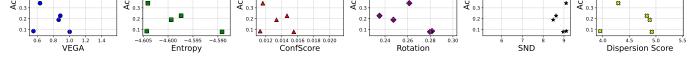| Dataset	ENT	Conf	Rot	SND	DS	VEGA	ENT	Conf	Rot	SND	DS	VEGA	ENT	Conf	Rot	SND	DS	VEGA	ENT	Conf	Rot	SND	DS	VEGA	Oracle
	R_5						τ_5						τ						Top-1 Acc.						Oracle
Cifar100	0.40	0.40	0.40	0.20	0.40	0.60	-1.00	-1.00	-1.00	0.00	-1.00	1.00	0.19	0.40	0.03	-0.21	0.35	0.71	0.80	0.80	0.72	0.09	0.80	0.82	0.82
Country211	0.60	0.80	0.60	0.20	0.00	0.80	-1.00	-0.33	0.33	0.00	0.00	-0.67	0.41	0.44	0.25	-0.22	0.24	0.51	0.29	0.29	0.32	0.03	0.01	0.29	0.33
DTD	0.20	0.20	0.60	0.40	0.20	0.40	1.00	0.67	0.00	0.00	0.67	0.67	0.59	0.68	-0.06	-0.24	0.56	0.72	0.68	0.68	0.51	0.10	0.68	0.68	0.68
Flowers	0.20	0.20	0.60	0.40	0.20	0.40	1.00	1.00	-1.00	-1.00	0.00	1.00	0.47	0.50	0.10	-0.25	0.26	0.68	0.69	0.69	0.73	0.07	0.55	0.88	0.88
GTSRB	0.20	0.20	0.60	0.40	0.40	0.60	-0.33	-0.33	1.00	0.00	0.00	0.33	0.43	0.49	0.18	0.06	0.46	0.65	0.48	0.48	0.40	0.07	0.48	0.64	0.64
MNIST	0.80	0.80	0.40	0.00	0.80	0.80	0.00	1.00	1.00	0.00	0.33	1.00	0.59	0.68	0.16	-0.53	0.59	0.59	0.77	0.88	0.65	0.08	0.81	0.88	0.88
Pets	0.40	0.60	0.40	0.20	0.40	0.80	-0.33	0.33	1.00	0.00	-0.33	0.67	0.51	0.61	0.10	-0.49	0.52	0.82	0.91	0.91	0.90	0.03	0.91	0.95	0.95
SVHN	0.60	0.60	0.60	0.00	0.20	0.80	0.00	-0.33	-1.00	0.00	1.00	0.67	0.59	0.53	0.18	-0.34	0.43	0.66	0.47	0.47	0.42	0.07	0.47	0.47	0.48
SUN397	0.40	0.60	0.40	0.20	0.40	0.60	0.67	0.60	0.00	0.00	0.33	0.67	0.38	0.65	-0.03	-0.31	0.43	0.82	0.04	0.75	0.61	0.19	0.53	0.75	0.75
Fer2013	0.20	0.20	0.60	0.20	0.40	0.60	1.00	1.00	-1.00	0.00	1.00	-1.00	0.24	0.21	0.07	-0.47	0.44	0.28	0.28	0.36	0.30	0.26	0.28	0.31	0.36
Avg.	0.40	0.46	0.52	0.22	0.34	0.64	0.10	0.26	-0.07	-0.10	0.23	0.46	0.44	0.52	0.10	-0.30	0.43	0.64	0.54	0.63	0.56	0.10	0.55	0.67	0.67

Fig. 4. Visualization of correlations between the actual zero-shot classification accuracy and the predicted scores for VLMs from various popular pre-training algorithms.
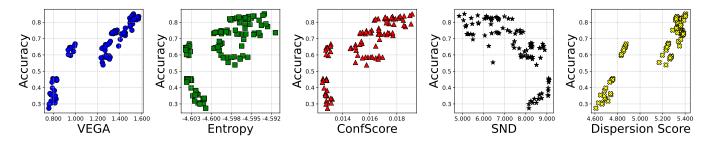

Fig. 5. Visualization of correlations between the actual downstream accuracy and the predicted scores across combinations of model and prompt template.

Fig. 4. VEGA exhibits a clear linear trend, and the DS points are also distributed along the diagonal. In contrast, the linear trends for other methods are less pronounced, especially Entropy and Confidence Score perform poorly in this setting.

C. Performance Prediction for Combinations of VLM and Prompt Template

In practical VLM usage, selecting both a suitable model and an appropriate prompt template is essential. Thus we conduct experiments to evaluate the performance of different combinations of templates and models. Note that Rotation [31] is not included in this comparison, as its calculation pertains only to the image encoder and does not account for variations in prompt templates. LOVM-G and LOVM-C are also as we explained in Sec. V-B.

Candidate Combinations. We select 10 CLIP models from open clip, including diverse network architectures and source datasets. The prompt templates are generated by GPT [60], with 10 different templates from simple to complex, short to long. There are a total of $10 \times 10 = 100$ combinations of model and template. Detailed information is provided in the supplementary material.

Quantitative Results. We compare the performance of different methods in predicting downstream results across various CLIP model and prompt template combinations, as shown in Table III . VEGA achieves the highest average R_5 of 0.36, τ of 0.56 and Top-1 accuracy of 0.58, demonstrating superior predictive accuracy across all downstream datasets. There are a lot of 0 results in τ_5, which is because the task is difficult, and the Top-5 recall (R_5) is low overall, so τ_5 is also low. Entropy performs well on this metric. Confidence Score also performs well overall, being the best on R_5 and second-best on the remaining three metrics. VEGA's consistent top performance across diverse datasets underscores its effectiveness in accurately predicting downstream performance for CLIP models and prompt template combinations.

Qualitative Results. Visualization of the correlations between actual downstream accuracy and predicted scores for comparison methods is shown in Fig. 5. The scatter plot for VEGA demonstrates a strong, positive linear correlation, with data points closely aligning along a diagonal line, indicating high predictive accuracy. In contrast, the plots for Entropy and SND show scattered patterns with no clear linear trend, reflecting weak correlations and lower predictive reliability. Confidence

TABLE III
DOWNSTREAM ZERO-SHOT CLASSIFICATION PERFORMANCE PREDICTION FOR COMBINATIONS OF CLIP MODELS AND PROMPT TEMPLATES.

Dataset	ENT	Conf	SND	DS	VEGA	ENT	Conf	SND	DS	VEGA	ENT	Conf	SND	DS	VEGA	ENT	Conf	SND	DS	VEGA	Oracle
	R_5					τ_5					τ					Top-1 Acc.					Oracle
Cifar100	0.40	0.60	0.00	0.80	0.20	1.00	1.00	0.00	0.00	0.00	0.46	0.61	-0.56	0.61	0.77	0.74	0.85	0.45	0.85	0.84	0.85
Country211	0.40	0.80	0.00	0.00	1.00	1.00	0.00	0.00	0.00	0.60	0.33	0.46	-0.35	0.30	0.47	0.22	0.28	0.17	0.18	0.28	0.28
DTD	0.40	0.60	0.00	0.00	0.60	1.00	1.00	0.00	0.00	-0.33	0.52	0.60	-0.34	0.55	0.73	0.65	0.65	0.44	0.57	0.64	0.65
Flowers	0.40	0.60	0.00	0.20	0.60	1.00	-0.82	0.00	-1.00	-0.33	0.49	0.56	-0.45	0.60	0.61	0.80	0.78	0.54	0.69	0.79	0.80
GTSRB	0.00	0.20	0.00	0.00	0.20	0.00	0.00	0.00	0.00	0.00	0.41	0.55	-0.30	0.45	0.56	0.43	0.48	0.29	0.38	0.48	0.51
MNIST	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.23	0.33	-0.20	0.42	0.37	0.46	0.37	0.41	0.61	0.46	0.71
Pets	0.80	0.80	0.00	0.00	0.80	0.00	-0.40	0.00	0.00	-0.60	0.60	0.64	-0.49	0.61	0.73	0.91	0.91	0.83	0.86	0.91	0.91
SVHN	0.00	0.00	0.00	0.00	0.20	0.00	0.00	0.00	0.00	0.00	0.42	0.48	-0.39	0.40	0.55	0.36	0.38	0.21	0.35	0.38	0.49
SUN397	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.46	0.60	-0.37	0.44	0.69	0.68	0.68	0.52	0.70	0.72	0.74
Fer2013	0.00	0.00	0.40	0.00	0.00	0.00	0.00	-1.00	0.00	0.00	-0.05	0.02	-0.07	0.05	0.11	0.18	0.18	0.31	0.21	0.32	0.35
Avg.	0.24	0.36	0.04	0.10	0.36	0.40	0.08	-0.10	-0.10	-0.07	0.39	0.48	-0.35	0.44	0.56	0.54	0.55	0.42	0.54	0.58	0.63

TABLE IV
ABLATION STUDY OF VEGA ON THREE BENCHMARKS MENTIONED ABOVE: PERFORMANCE PREDICTION FOR (A) VLMs FROM CLIP FAMILY (SEC. V-A); (B) VLMs FROM VARIOUS PRE-TRAINING ALGORITHMS (SEC. V-B); AND (C) COMBINATIONS OF VLM AND PROMPT TEMPLATE (SEC. V-C). EACH ROW REPRESENTS THE AVERAGE RESULTS ON THESE BENCHMARKS.

	s_n (Node)				s_e (Edge)				$s_n + s_e$ (VEGA)			
	R_5	τ_5	τ	Top-1 Acc.	R_5	τ_5	τ	Top-1 Acc.	R_5	τ_5	τ	Top-1 Acc.
(a)	0.62	0.49	0.60	0.62	0.44	-0.13	0.44	0.61	0.64	0.20	0.62	0.62
(b)	0.68	0.39	0.56	0.67	0.68	0.01	0.01	0.63	0.64	0.46	0.64	0.67
(c)	0.42	0.03	0.55	0.63	0.16	0.00	0.55	0.63	0.36	-0.07	0.56	0.63

Score and Dispersion Score exhibit moderate correlations with some linearity, but their data points are more dispersed compared to VEGA.

D. Ablation Study

Table IV presents the ablation study of VEGA on three benchmarks mentioned in the above sections: (a) prediction on VLMs from the CLIP family (Sec. V-A), (b) prediction on VLMs from various pre-training algorithms (Sec. V-B), and (c) prediction on combinations of VLMs and prompt templates (Sec. V-C). The study investigates the contributions of node similarity s_n and edge similarity s_e individually, as well as their combination $s_n + s_e$ which constitutes the full VEGA method. In all cases, the full VEGA method, which combines both node and edge similarity, achieves the highest predictive accuracy with R^2 and ρ values surpassing those of using s_n or s_e individually. For the CLIP family benchmark (a), VEGA achieves the best in three of the four metrics, indicating its strong predictive capability. Similar trends are observed in the other two benchmarks, with VEGA outperforming its individual components, highlighting the robustness and effectiveness of integrating both node and edge similarities for downstream performance prediction. The role of node similarity is greater than that of edge, and the combination of edge and node can improve τ, indicating that the sum of node and edge similarity can more comprehensively evaluate the performance of the model.

E. Sensitive Analysis

In the calculation of node similarity in Eq. (12), we introduce a temperature parameter t to sharpen the node score s_n. The value of t is empirically set to 0.05 and is kept consistent across all experiments, including different models and downstream datasets. In this section, we provide a sensitivity analysis of t on the prediction for VLMs from CLIP family. For each dataset, the figure presents the values of four metrics for five different temperature settings around the default $t=0.05$, ranging from $t = 0.005$ to $t = 0.5$. We report the average results on ten downstream datasets in Fig. 6. The results indicate that VEGA maintains stable performance across varying temperatures.
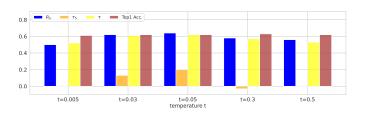

Fig. 6. Sensitivity analysis on temperature t in Eq. (12). The Y-axis is the average results of the prediction for VLMs from CLIP family (Sec V-A).

VI. CONCLUSION

This paper introduces a novel method called Visual-tExtual Graph Alignment (VEGA) for unsupervised vision language model selection, without access to downstream dataset annotations or the training data of VLMs. The core intuition behind VEGA is that models with similar structures in textual and visual features are more effective at matching images with their corresponding labels. Specifically, we construct two fully connected graphs representing the class distributions for visual and textual modalities, and define the VEGA score as the similarity between these two graphs. We establish three benchmarks across practical application scenarios for VLM performance prediction. VEGA outperforms existing baselines, demonstrating its effectiveness and reliability in

estimating VLM performance for unlabeled downstream tasks, and the generalizability in various scenarios. We hope this work provides valuable insights for further research in this field.

REFERENCES

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *Proc. ICML*, 2021, pp. 8748–8763.

[2] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig, "Scaling up visual and vision-language representation learning with noisy text supervision," in *Proc. ICML*, 2021, pp. 4904–4916.

[3] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proc. ICCV*, 2023, pp. 11 975–11 986.

[4] F. Peng, X. Yang, L. Xiao, Y. Wang, and C. Xu, "Sgva-clip: Semantic-guided visual adapting of vision-language models for few-shot image classification," *IEEE Transactions on Multimedia*, vol. 26, pp. 3469–3480, 2023.

[5] L. Xiao, X. Yang, F. Peng, M. Yan, Y. Wang, and C. Xu, "Clip-vg: Self-paced curriculum adapting of clip for visual grounding," *IEEE Transactions on Multimedia*, vol. 26, pp. 4334–4347, 2023.

[6] X. Yang, F. Liu, and G. Lin, "Effective end-to-end vision language pre-training with semantic visual loss," *IEEE Transactions on Multimedia*, vol. 25, pp. 8408–8417, 2023.

[7] ——, "Neural logic vision language explainer," *IEEE Transactions on Multimedia*, vol. 26, pp. 3331–3340, 2024.

[8] O. Zohar, S.-C. Huang, K.-C. Wang, and S. Yeung, "Lovm: Language-only vision model selection," in *Proc. NeurIPS Workshops*, 2024.

[9] C. Yi, D.-C. Zhan, and H.-J. Ye, "Bridge the modality and capacity gaps in vision-language model selection," *arXiv preprint arXiv:2403.13797*, 2024.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, pp. 211–252, 2015.

[11] OpenAI, "Gpt-4 technical report," in *arXiv preprint arXiv:2303.08774*, 2024.

[12] S. Garg, S. Balakrishnan, Z. C. Lipton, B. Neyshabur, and H. Sedghi, "Leveraging unlabeled data to predict out-of-distribution performance," in *Proc. ICLR*, 2022.

[13] W. Deng and L. Zheng, "Are labels always necessary for classifier accuracy evaluation?" in *Proc. CVPR*, 2021, pp. 15 069–15 078.

[14] Y. Yu, Z. Yang, A. Wei, Y. Ma, and J. Steinhardt, "Predicting out-of-distribution error with the projection norm," in *Proc. ICML*, 2022, pp. 25 721–25 746.

[15] A. Fang, G. Ilharco, M. Wortsman, Y. Wan, V. Shankar, A. Dave, and L. Schmidt, "Data determines distributional robustness in contrastive language image pre-training (clip)," in *Proc. ICML*, 2022, pp. 6216–6234.

[16] P. Mayilvahanan, T. Wiedemer, E. Rusak, M. Bethge, and W. Brendel, "Does clip's generalization performance mainly stem from high train-test similarity?" in *Proc. ICLR*, 2024.

[17] H. Yu, J. Liu, X. Zhang, J. Wu, and P. Cui, "A survey on evaluation of out-of-distribution generalization," in *arXiv preprint arXiv:2403.01874*, 2024.

[18] Y. Ding, B. Jiang, A. Yu, A. Zheng, and J. Liang, "Which model to transfer? a survey on transferability estimation," *arXiv preprint arXiv:2402.15231*, 2024.

[19] Q. Garrido, R. Balestriero, L. Najman, and Y. Lecun, "Rankme: Assessing the downstream performance of pretrained self-supervised representations by their rank," in *Proc. ICML*, 2023, pp. 10 929–10 974.

[20] L.-Z. Guo, Z. Zhou, Y.-F. Li, and Z.-H. Zhou, "Identifying useful learnwares for heterogeneous label spaces," in *Proc. ICML*, 2023, pp. 12 122–12 131.

[21] M. Gholami, M. Akbari, X. Wang, B. Kamranian, and Y. Zhang, "Etran: Energy-based transferability estimation," in *Proc. ICCV*, 2023, pp. 18 613–18 622.

[22] Y. Lu, Z. Wang, R. Zhai, S. Kolouri, J. Campbell, and K. Sycara, "Predicting out-of-distribution error with confidence optimal transport," in *Proc. NeurIPS*, 2023.

[23] F. Mosteller and J. W. Tukey, "Data analysis and regression. a second course in statistics," *Addison-Wesley series in behavioral science: quantitative methods*, 1977.

[24] K. Saito, D. Kim, P. Teterwak, S. Sclaroff, T. Darrell, and K. Saenko, "Tune it the right way: Unsupervised validation of domain adaptation via soft neighborhood density," in *Proc. ICCV*, 2021, pp. 9184–9193.

[25] M. Sugiyama, M. Krauledat, and K.-R. Müller, "Covariate shift adaptation by importance weighted cross validation." *Journal of Machine Learning Research*, vol. 8, no. 5, 2007.

[26] K. You, X. Wang, M. Long, and M. Jordan, "Towards accurate model selection in deep unsupervised domain adaptation," in *Proc. ICML*, 2019, pp. 7124–7133.

[27] H. Namkoong and J. C. Duchi, "Stochastic gradient methods for distributionally robust optimization with f-divergences," in *Proc. NeurIPS*, 2016.

[28] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant risk minimization," in *arXiv preprint arXiv:1907.02893*, 2019.

[29] Z. Shen, P. Cui, T. Zhang, and K. Kunag, "Stable learning via sample reweighting," in *Proc. AAAI*, 2020, pp. 5692–5699.

[30] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *Proc. ICLR*, 2017.

[31] W. Deng, S. Gould, and L. Zheng, "What does rotation prediction tell us about classifier accuracy under varying testing environments?" in *Proc. ICML*, 2021, pp. 2579–2589.

[32] C. Baek, Y. Jiang, A. Raghunathan, and J. Z. Kolter, "Agreement-on-the-line: Predicting the performance of neural networks under distribution shift," in *Proc. NeurIPS*, 2022, pp. 19 274–19 289.

[33] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *Proc. ICML*, 2022, pp. 12 888–12 900.

[34] K. Pearson, "I. mathematical contributions to the theory of evolution.—vii. on the correlation of characters not quantitatively measurable," *Philosophical Transactions of the Royal Society of London.*, vol. 195, no. 262-273, pp. 1–47, 1900.

[35] A. Krizhevsky, "Learning multiple layers of features from tiny images," *Master's thesis, University of Tront*, 2009.

[36] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar, "Cats and dogs," in *Proc. CVPR*, 2012, pp. 3498–3505.

[37] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. ICVGIP*, 2008, pp. 722–729.

[38] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Y. Ng *et al.*, "Reading digits in natural images with unsupervised feature learning," in *Proc. NeurIPS Workshops*, 2011.

[39] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The german traffic sign detection benchmark," in *Proc. IJCNN*, 2013, pp. 1–8.

[40] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. CVPR*, 2014, pp. 3606–3613.

[41] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "Yfcc100m: The new data in multimedia research," *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

[42] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *Proc. CVPR*, 2010, pp. 3485–3492.

[43] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. IEEE*, 1998, pp. 2278–2324.

[44] Dumitru, G. Ian, C. Will, and B. Yoshua, "Challenges in representation learning: Facial expression recognition challenge," 2013. [Online]. Available: https://kaggle.com/competitions/challenges-in-representation-learning-facial-expression-recognition-challenge

[45] R. Xie, H. Wei, L. Feng, Y. Cao, and B. An, "On the importance of feature separability in predicting out-of-distribution error," in *Proc. NeurIPS*, 2023.

[46] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. ICLR*, 2018.

[47] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[48] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[50] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2021.

[51] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," in *Proc. CVPR*, 2022, pp. 11 976–11 986.

[52] C. Schuhmann, R. Vencu, R. Beaumont, R. Kaczmarczyk, C. Mullis, A. Katta, T. Coombes, J. Jitsev, and A. Komatsuzaki, "Laion-400m: Open dataset of clip-filtered 400 million image-text pairs," in *Proc. NeurIPS Workshops*, 2021.

[53] Z. Chen, G. Liu, B.-W. Zhang, Q. Yang, and L. Wu, "AltCLIP: Altering the language encoder in CLIP for extended language capabilities," in *Proc. ACL*, 2023, pp. 8666–8682.

[54] J. Xu, S. De Mello, S. Liu, W. Byeon, T. Breuel, J. Kautz, and X. Wang, "Groupvit: Semantic segmentation emerges from text supervision," in *Proc. CVPR*, 2022, pp. 18 134–18 144.

[55] L. Haas, S. Alberti, and M. Skreta, "Learning generalized zero-shot learners for open-domain image geolocalization," in *arXiv preprint arXiv:2302.00275*, 2023.

[56] H. Xu, S. Xie, X. E. Tan, P.-Y. Huang, R. Howes, V. Sharma, S.-W. Li, G. Ghosh, L. Zettlemoyer, and C. Feichtenhofer, "Demystifying clip data," in *arXiv preprint arXiv:2309.16671*, 2023.

[57] S. Zhang, Y. Xu, N. Usuyama, J. Bagga, R. Tinn, S. Preston, R. Rao, M. Wei, N. Valluri, C. Wong, M. Lungren, T. Naumann, and H. Poon, "Large-scale domain-specific pretraining for biomedical vision-language processing," in *arXiv preprint arXiv:2303.00915*, 2023.

[58] W. O. Ikezogwo, M. S. Seyfioglu, F. Ghezloo, D. S. C. Geva, F. S. Mohammed, P. K. Anand, R. Krishna, and L. Shapiro, "Quilt-1m: One million image-text pairs for histopathology," in *arXiv preprint arXiv:2306.11207*, 2023.

[59] S. Stevens, J. Wu, M. J. Thompson, E. G. Campolongo, C. H. Song, D. E. Carlyn, L. Dong, W. M. Dahdul, C. Stewart, T. Berger-Wolf, W.-L. Chao, and Y. Su, "BioCLIP: A vision foundation model for the tree of life," in *Proc. CVPR*, 2024.

[60] OpenAI, "Gpt-4: Generative pre-trained transformer," https://openai.com/gpt-4, 2023, accessed: 2024-08-14.