Inferring Latent Class Statistics from Text for Robust Visual Few-Shot Learning

Yassir Bendou
IMT Atlantique, Brest, France
yassir.bendou@imt-atlantique.fr

Vincent Gripon
IMT Atlantique, Brest, France
vincent.gripon@imt-atlantique.fr

Bastien Pasdeloup
IMT Atlantique, Brest, France
bastien.pasdeloup@imt-atlantique.fr

Giulia Lioi
IMT Atlantique, Brest, France
giulia.lioi@imt-atlantique.fr

Lukas Mauch
Sony Europe, B.V.
Stuttgart Laboratory 1, Germany
lukas.mauch@sony.com

Fabien Cardinaux
Sony Europe, B.V.
Stuttgart Laboratory 1, Germany
fabien.cardinaux@sony.com

Ghouthi Boukli Hacene
Sony Europe, B.V.
Stuttgart Laboratory 1, Germany
Mila, Montreal, Canada
ghouthi.bouklihacene@sony.com

Abstract

In the realm of few-shot learning, foundation models like CLIP have proven effective but exhibit limitations in cross-domain robustness especially in few-shot settings. Recent works add text as an extra modality to enhance the performance of these models. Most of these approaches treat text as an auxiliary modality without fully exploring its potential to elucidate the underlying class visual features distribution. In this paper, we present a novel approach that leverages text-derived statistics to predict the mean and covariance of the visual feature distribution for each class. This predictive framework enriches the latent space, yielding more robust and generalizable few-shot learning models. We demonstrate the efficacy of incorporating both mean and covariance statistics in improving few-shot classification performance across various datasets. Our method shows that we can use text to predict the mean and covariance of the distribution offering promising improvements in few-shot learning scenarios.

1 Introduction

Few-shot learning has experienced a significant transformation due to the advent of foundation models. These large pre-trained vision models, such as CLIP [1] and DINO [2] have shown remarkable performance in various applications but still face challenges in terms of robustness and performance consistency in cross-domain settings. Our work aims to address these limitations by exploring the synergistic use of text as an additional modality.

Text serves as a rich source of semantic information that can offer insights into the class definition, clarify concepts, and even outline the boundaries of what a class entails. While some previous works

R0-FoMo: Workshop on Robustness of Few-shot and Zero-shot Learning in Foundation Models at NeurIPS 2023.

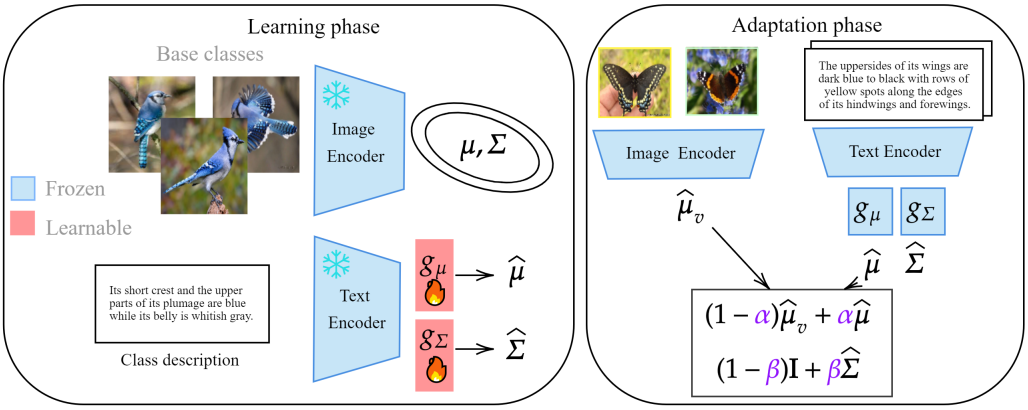


Figure 1: Our pipeline is composed of two phases. We first train two networks to infer the mean and the covariance of the latent visual distribution. Secondly, given a few-shot downstream task, we interpolate the predictions of the mean using the available shots and apply shrinkage for the covariance matrix. We then use these statistics to perform few-shot classification.

have added text as a supplemental feature for classification, we aim at proposing a method exploring more nuanced ways of aligning text and images to fully exploit semantic knowledge.

Our primary focus is to examine whether statistics such as the covariance of the distribution of visual features for a class can be inferred from text. This is especially pertinent in few-shot learning scenarios, where we often lack comprehensive data to understand the class distribution fully. By being able to predict statistics about the feature distributions from text, we can potentially improve the robustness and generalizability of few-shot classifiers. Specifically, we explore:

- If we can predict the mean and covariance of visual feature distribution from text.
- The utility of such predictions in enhancing the performance of few-shot learning models.

By focusing on these questions, our work explores the next frontier in few-shot learning, pushing the boundaries of what is achievable with current foundation models by augmenting them with rich, text-derived statistical information. The code for our experiments is available at: `https://github.com/ybendou/fs-text2stats`.

2 Related Work

Multimodal foundation models In recent years, foundation models have increasingly adopted a multimodal supervision paradigm, integrating multiple types of data such as text and images for more effective learning. In the domain of vision, initial research efforts focused on converting web-based image captions into structured formats suitable for supervised learning. These formats included multi-label classification targets and visual n-grams. More contemporary methods, exemplified by CLIP and ALIGN [3], employ a contrastive learning framework to map both images and their associated captions into a shared representational space. These methods have shown remarkable capabilities in "zero-shot" learning, demonstrating high performance on downstream tasks without requiring additional fine-tuning.

Text for visual few-shot classification Several approaches have benefited from the extensive pre-training provided by models like CLIP, which utilizes a large-scale dataset of image-caption pairs to train robust encoders for both text and images [1, 4, 5]. There are many streams of work to adapt text to visual tasks. The first one is prompt learning which aims to learn effective prompts in the presence of limited labeled data [6]. These approaches differ in how they optimize for prompt generalization across tasks and classes. For instance, CoCoOp introduces query-based token conditioning for better generalizability [7], while PLOT employs an optimal transport framework to combine multiple local prompts [8]. Another strand of work, including WiSE-FT and Clip-Adapter, focuses on optimizing the fine-tuning process by introducing lightweight, additive modules to the pre-trained models. These methods vary in the extent to which they leverage the limited shots available for fine-tuning [9–11].

Recent work has examined how text can function as additional data points for training [12]. Methods such as DALLE-based shot generation go beyond mere text prompts, employing generated images as additional training samples for classification [13]. Another work trains a VAE conditioned on semantic embeddings to generate additional points in the feature space [14]. Unlike these approaches, our work considers text as a statistical summary of the class, capturing both mean and covariance.

While previous work has mostly relied on simplistic prompts based solely on class labels, some recent studies have demonstrated the utility of more descriptive class prompts. Our approach extends this line of work by experimenting with both class labels and more informative visual class descriptions [15].

3 Approach

In this section, we present our method to predict from text the mean and covariance of a class distribution in the feature space. We split our method in two phases, a learning phase where we learn a mapping from text to these order of moments, and an adaptation phase where we adjust the predictions to the downstream dataset.

3.1 Learning Phase

In the learning phase, our objective is to predict both the mean and covariance of visual features using textual descriptions. Having access to statistics about the visual distribution are important for few-shot learning where only few samples are available to infer the latent distribution. Let $\mathbf{x}$ be an image and $\mathbf{t}$ the text of its associated label y. We aim to utilize a pre-trained visual backbone, denoted by f_v, to extract image features $\mathbf{z} = f_v(\mathbf{x})$ and a pre-trained text encoder, denoted by f_t, to extract text features $\mathbf{s} = f_t(\mathbf{t})$. For text, we either utilize different text contexts such as "a photo of a {class}" or we use GPT3 [16] to generate a visual description for each class, which has been used previously in the literature [13] (see Appendix for more details).

Given some semantic features $\mathbf{s}_i$ of a class c_i, our target is to accurately estimate the mean and covariance of the visual features $\boldsymbol{\mu}_i = \mathbb{E}_{\substack{(\mathbf{x},y)\\ y=c_i}}(f_v(\mathbf{x}))$ and $\boldsymbol{\Sigma}_i = \mathbb{E}_{\substack{(\mathbf{x},y)\\ y=c_i}}\left((f_v(\mathbf{x}) - \mu_i)(f_v(\mathbf{x}) - \mu_i)^T\right)$. Given the high-dimensionality of these features, we choose to infer a diagonal covariance matrix.

We employ two mapping networks, $g_\mu(\mathbf{s}, \theta_\mu)$ and $g_\Sigma(\mathbf{s}, \theta_\Sigma)$ for predicting the mean and covariance. Leaning on the strength of CLIP for robust textual embeddings, these networks are compactly designed, primarily using a Multi-Layer Perceptrons with 2 layers. The training for these networks is executed on a base dataset (more details in Section 4). Using the base dataset composed of N classes, we compute the empirical mean and covariance for each class $\tilde{\boldsymbol{\mu}}_i$ and $\tilde{\boldsymbol{\Sigma}}_i$ respectively and train the mapping networks using an $L2$ loss on these statistics combined with a regularization term R (*e.g.*, weight decay): $L(\theta_{\boldsymbol{\mu}}) = \frac{1}{N}\sum_{i=1}^{N} \|g_\mu(\mathbf{s}_i, \theta_\mu) - \tilde{\boldsymbol{\mu}}_i\|_2 + R(\boldsymbol{\theta}_{\boldsymbol{\mu}})$ and $L(\theta_{\boldsymbol{\Sigma}}) = \frac{1}{N}\sum_{i=1}^{N} \|g_\Sigma(\mathbf{s}_i, \theta_\Sigma) - \tilde{\boldsymbol{\Sigma}}_i\|_2 + R(\boldsymbol{\theta}_{\boldsymbol{\Sigma}})$.

3.2 Adaptation Phase

Having inferred estimators for the mean and the covariance, integrating them into the classification process would be straightforward, as delineated in Section 4. Nevertheless, to account for potential domain shifts between training and test datasets, we introduce regularization techniques. Specifically for the mean, similar to [17], we interpolate our estimate using the empirical mean from the few available shots in the downstream task denoted $\hat{\boldsymbol{\mu}}_{i,v}$ as follows: $\hat{\boldsymbol{\mu}}_i = (1-\alpha)\,\hat{\boldsymbol{\mu}}_{i,v} + \alpha g_\mu(\boldsymbol{s}_i)$. Due to the unreliability of covariance estimates from few-shots, we employ shrinkage, adjusted by a coefficient β: $\hat{\boldsymbol{\Sigma}}_i = (1-\beta)\,\mathbf{I} + \beta g_\Sigma(\boldsymbol{s}_i)$. In our experiments, we assume that the covariance is diagonal, which means that a simple interpolation in the euclidean space is sufficient.

4 Experiments

4.1 Experimental protocol

In our experiments, we use two base datasets: ImageNet [18] and iNaturalist [19]. We chose iNaturalist given that it's a hierchical dataset with fine-grained classes (different species that are very

similar to each other sometimes), in such a dataset, the covariance matrix plays a crucial role. For the test dataset, we experiment with either a disjoint subset of the base classes or 9 cross-domain datasets.: Caltech [20], EuroSAT [21], Food [22], Flowers [23], SUN397 [24], DTD [25], Pets [26], Cars [27] and UCF101 [28]. We extract visual and text features using the pre-trained CLIP ResNet50 which was trained on LAION400M [29], a disjoint dataset from the previously mentionned ones.

4.2 Mapping networks training

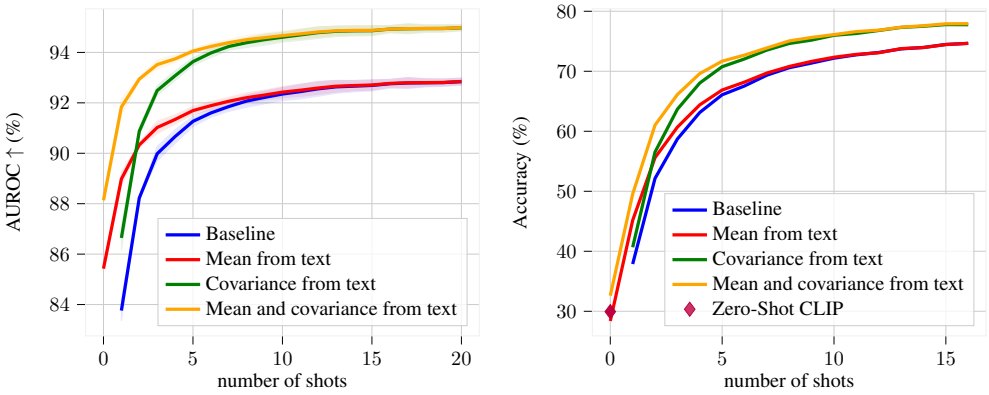
Base dataset		Model	$\boldsymbol{\mu}$ val	$\boldsymbol{\mu}$ test	$\boldsymbol{\Sigma}$ val	$\boldsymbol{\Sigma}$ test
iNaturalist	ID	Baseline	1.00 ± 0.02	0.97 ± 0.02	1.00 ± 0.03	0.99 ± 0.03
		Class labels	0.60 ± 0.02	0.60 ± 0.02	0.66 ± 0.03	0.72 ± 0.03
		Class description	$\mathbf{0.42 \pm 0.01}$	$\mathbf{0.42 \pm 0.01}$	$\mathbf{0.54 \pm 0.02}$	$\mathbf{0.56 \pm 0.02}$
ImageNet	ID	Baseline	1.00 ± 0.02	0.98 ± 0.02	1.01 ± 0.05	1.00 ± 0.04
		Class labels	$\mathbf{0.44 \pm 0.02}$	$\mathbf{0.45 \pm 0.01}$	$\mathbf{0.70 \pm 0.03}$	$\mathbf{0.74 \pm 0.04}$
		Class description	0.46 ± 0.02	0.47 ± 0.01	$\mathbf{0.70 \pm 0.03}$	$\mathbf{0.74 \pm 0.04}$
	CD	Baseline	0.99 ± 0.03	1.67 ± 0.10	1.04 ± 0.07	4.99 ± 0.90
		Class labels	$\mathbf{0.43 \pm 0.03}$	$\mathbf{1.26 \pm 0.11}$	0.72 ± 0.06	$\mathbf{3.99 \pm 0.60}$
		Class description	0.45 ± 0.02	1.30 ± 0.11	$\mathbf{0.70 \pm 0.05}$	4.67 ± 0.84

Table 1: Mean Squared Error of predictions from ImageNet-1k and iNaturalist. The baseline is the average value over the training set. We experiment with both in-domain (ID) and cross-domain (CD) settings. The results are averaged across 10 different seeds.

Let us first show that it is possible to accurately estimate both means and covariances of visual features of class from its label. To this end, we train a model to predict the mean (resp. the covariance) of visual features from the text features of its class label. These models were trained to minimize the MSE on a train set, then tested on a validation set, either in a In-Domain (ID) setting or Cross-Domain (CD) setting. Means and covariances were standardized for this experiment, such that the MSE of the baseline in the ID case is 1 (more details in the Appendix).

We can derive multiple results from this first experiment in Table 1. First, we observe that in the ID case, our models obtain MSE significantly lower than 1, showing there is indeed information about both the mean and covariance hidden in the text model. Second, we observe that for iNaturalist, using the description of a class gives significantly better results than the class labels which is expected given that iNaturalist often contains uncommon class names. This result does not hold for ImageNet which contains common names. Therefore, for all our experiments, we use class descriptions for iNaturalist and class labels when using ImageNet as a base dataset.

4.3 Classification



(a) One-class (b) Multi-class

Figure 2: Results on iNaturalist Dataset for both one-class and multi-class classification tasks using 10^3 few-shot runs. For all the models we use a Mahalanobis classifier. We compare the baseline without text to three models.

Next, we use the mean and covariance predictions from text for few-shot classification tasks and show that these bring consistent improvements over the existing baselines in both one-class and multi-class scenarios. We start from a baseline which uses only the few available shots with no text supervision i.e. $\alpha = 0$ and $\beta = 0$. We compare the baseline to three models: M (mean from the shots and text) i.e. $\beta = 0$, C (covariance from the text) i.e. $\alpha = 0$ and M&C (mean and covariance from text). For the multi-class setting, we select the best α and β coefficient using a held-out validation set with $min(n, 4)$ examples, where n is the number of shots, which is a common practice in the field [6].

4.3.1 One-class classification

Let us first consider the one-class classification case. In this setting, the goal is to determine if test samples belong to a specific class or not. It contrasts with multi-class. In this context, the covariance matrix becomes pivotal. Using the Mahalanobis distance, the sample likelihood is:

$$p(x|\mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right) \tag{1}$$

We report the AUROC (Area Under the ROC Curve) score for different methods in Figure 2 for iNaturalist in-domain and Table 2 for cross-domain datasets. For each run, we select a random class in the test set with k random shots from the corresponding class and select with a probability of 0.5 a query either from the same class or from another class. For iNaturalist, our method shows a consistent improvement over the baseline. We show that using the mean from text brings an improvement in the very low shot regime ($k \leq 10-$shots). While including the covariance brings a consistent improvement of $\simeq 2\%$ for all data regimes. Particularly in a very low shot regime, it's interesting to combine both mean and covariance predictions from text which brings over 8% AUROC improvement. Furthermore, our method allows to perform zero-shot with performances equivalent to $2-$shots classification.

In the cross-domain setting, Table 2 shows that for most of the datasets, our method (M&C) brings a significant improvement over the baseline. Overall, using the covariance seems to bring more improvement than using the mean. This could be explained by the importance of the covariance for the one-class classification task. Similarly to iNaturalist, using the mean helps more in the very low shot regimes and the using covariance helps for all the different number of shots that we tested.

	Method	Shots	Dataset									
			Caltech	EuroSAT	Food	Flowers	SUN397	DTD	Pets	Cars	UCF101	Average
AUROC	Baseline	1	89.74	81.34	83.34	90.92	89.18	71.23	81.47	90.31	84.70	84.70
		2	93.31	85.79	87.59	94.66	94.37	76.36	86.16	94.28	88.63	89.02
		4	94.95	88.60	89.54	96.38	96.47	79.53	88.11	96.04	90.70	91.15
		8	95.89	89.85	90.36	97.09	97.51	80.88	88.83	96.78	91.72	92.10
		16	96.49	90.39	90.79	97.51	97.83	81.63	89.21	97.12	92.25	92.57
Δ%	Mean from text (M)	1	+2.07	0.00	+0.68	+0.48	+2.83	+0.17	+6.31	+0.38	+1.37	1.59
		2	+0.46	0.00	0.00	0.00	+0.22	0.00	+2.17	0.00	+0.11	0.33
		4	+0.10	0.00	0.00	0.00	0.00	0.00	+0.76	0.00	0.00	0.10
		8	0.00	0.00	0.00	0.00	0.00	0.00	+0.37	0.00	0.00	0.04
		16	0.00	0.00	0.00	0.00	0.00	0.00	+0.18	0.00	0.00	0.02
	Covariance from text (C)	1	+2.09	+1.47	+3.05	+3.59	+0.08	0.00	+4.57	+1.97	0.00	+1.86
		2	+1.80	+1.45	+3.09	+2.53	+0.21	0.00	+3.38	+1.66	0.00	+1.57
		4	**+1.49**	**+1.27**	**+2.92**	**+1.82**	**+0.21**	**0.00**	**+2.76**	**+1.23**	**0.00**	**+1.30**
		8	**+1.15**	**+1.22**	**+2.73**	**+1.45**	**+0.22**	**0.00**	**+2.53**	**+1.07**	**0.00**	**+1.15**
		16	**+0.96**	**+1.20**	**+2.64**	**+1.16**	**+0.25**	**0.00**	**+2.40**	**+1.00**	**0.00**	**+1.07**
	Mean and covariance from text (M&C)	1	**+3.82**	**+1.47**	**+3.61**	**+3.99**	**+2.85**	**+0.17**	**+8.43**	**+2.33**	**+1.37**	**+3.11**
		2	**+2.07**	**+1.45**	**+3.09**	**+2.54**	**+0.34**	0.00	**+4.18**	**+1.66**	**+0.11**	**+1.71**
		4	**+1.52**	**+1.27**	**+2.92**	**+1.82**	**+0.21**	0.00	**+2.79**	**+1.23**	+0.01	**+1.31**
		8	**+1.15**	**+1.22**	**+2.73**	**+1.45**	**+0.22**	0.00	**+2.53**	**+1.07**	0.00	**+1.15**
		16	**+0.96**	**+1.20**	**+2.64**	**+1.16**	**+0.25**	0.00	**+2.40**	**+1.00**	0.00	**+1.07**

Table 2: Results per dataset on one-class tasks. We compute the AUROC score over 100 tasks and average the performance over 20 runs. We report the score difference with respect to the Baseline for different number of shots. We highlight the best performance gains compared to the baseline. The confidence intervals range from 0.07% to 0.67% which we omitted for better readability.

4.3.2 Multi-class classification

In the multi-class scenario, our approach utilizes the same Mahanalobis classifier, a generalization of the Nearest Class Mean which has been widely adopted in the few-shot literature [30–32]:

$$p(y = c/x) = \frac{\exp\left(\sqrt{\left(\hat{\mu}_c - \mathbf{x}\right)^T \hat{\mathbf{\Sigma}}_c^{-1} \left(\hat{\mu}_c - \mathbf{x}\right)}\right)}{\sum_{c'} \exp\left(\sqrt{\left(\hat{\mu}_{c'} - \mathbf{x}\right)^T \hat{\mathbf{\Sigma}}_{c'}^{-1} \left(\hat{\mu}_{c'} - \mathbf{x}\right)}\right)} \tag{2}$$

For this setting, we report the performance over 10^3 runs. Similarly to the one-class setting, our baseline benefits from the additional gain from the mean and the covariance. On iNaturalist, the gain in multi-class is smaller, which could be explained by the difficulty of the task and the fact that one-class depends strongly on a good estimate of the boundaries of the distribution which is reflected in the covariance matrix. Furthermore, we observe on Table 3 that the benefits of our methods are particularly important on datasets where the baseline is low compared to the zero-shot performance (Pets and SUN397). In the case where the zero-shot is lower than the baseline, our method still brings improvements, for example we can observe on EuroSAT 2% increase for all shot settings.

	Method	Shots	Dataset									
			Caltech	EuroSAT	Food	Flowers	SUN397	DTD	Pets	Cars	UCF101	Average
	Zero-Shot CLIP	0	82.39	36.01	75.72	62.28	56.60	40.01	79.59	51.91	59.42	60.43
Accuracy	Baseline	1	62.07	47.80	27.55	52.25	28.15	26.11	26.69	21.62	36.89	36.57
		2	74.95	58.30	40.33	69.10	41.42	37.64	38.28	33.18	50.08	49.25
		4	81.31	66.07	51.67	80.49	51.89	47.49	49.65	44.02	58.80	59.04
		8	84.61	70.67	59.72	86.52	58.59	54.17	58.63	52.53	64.23	65.52
		16	86.34	73.49	64.68	89.59	62.46	58.21	65.23	65.18	67.48	70.68
Δ%	Mean from text (M)	1	+16.46	+0.82	+13.80	+6.58	+17.22	+6.94	+26.56	+5.26	+9.98	+11.51
		2	+6.30	+0.33	+8.09	+2.58	+9.00	+3.62	+19.74	+2.04	+4.06	+06.20
		4	+2.32	-0.07	+3.85	+0.75	+3.66	+1.49	+13.40	+0.58	+1.53	+03.06
		8	+0.76	-0.19	+1.57	+0.20	+1.30	+0.55	+8.91	+0.13	+0.51	+01.53
		16	+0.21	-0.33	+0.67	+0.04	+0.43	+0.28	+5.61	+5.76	+0.15	+0.91
	Covariance from text (C)	1	+1.56	+0.84	+4.06	+6.34	+0.74	+1.79	+6.64	+4.93	+1.48	+03.15
		2	+2.92	+1.52	+6.72	+6.84	+2.16	+3.75	+11.10	+7.58	+3.50	+05.12
		4	+2.85	+1.97	+7.41	+5.35	+3.21	+5.27	+13.29	+8.72	+4.55	+05.85
		8	+2.44	+2.03	+6.71	+3.88	+3.59	+5.44	+12.60	+8.77	+4.98	+05.60
		16	+2.29	+1.99	+5.80	+2.95	+3.59	+5.33	+11.31	+11.22	+5.11	+04.82
	Mean and covariance from text (M&C)	1	**+21.03**	**+2.20**	**+25.57**	**+14.90**	**+22.14**	**+14.05**	**+37.86**	**+16.14**	**+14.94**	**+18.76**
		2	**+10.27**	**+2.04**	**+18.42**	**+9.82**	**+13.74**	**+10.45**	**+31.86**	**+11.22**	**+9.48**	**+13.03**
		4	**+5.76**	**+2.21**	**+12.49**	**+6.21**	**+8.23**	**+8.15**	**+25.80**	**+9.67**	**+7.27**	**+09.53**
		8	**+3.71**	**+2.09**	**+8.78**	**+4.21**	**+5.56**	**+6.71**	**+20.42**	**+9.06**	**+6.29**	**+07.43**
		16	**+2.81**	**+2.02**	**+6.74**	**+3.13**	**+4.46**	**+6.10**	**+16.30**	**+15.99**	**+5.91**	**+5.94**

Table 3: Results per dataset. We report the performance across 10^2 tasks of Zero-shot CLIP and the baseline, for other methods we report the average difference to the baseline. We observe that combining and mean and covariance (M&C) consistently improves the baseline. The confidence intervals range from 0.09% to 0.35% which we omitted for better readability.

5 Discussion

Let us now summarize and discuss our findings through Q&As.

Q1: Can we predict the mean and the covariance from text? **A**: Yes, our experiment shows that we are able to predict the mean and covariance. Our interpretation is that the multi-modal training of CLIP helps aligning the two modalities, therefore training a small MLP network is sufficient to learn the mapping to the mean and covariance.

Q2: Does Covariance bring improvement in few-shot classification? **A**: Yes, we show that for both one-class and multi-class settings, using the covariance matrix to normalize the data brings an improvement. Especially when dealing with fine-grained concepts such as species from iNaturalist.

Q3: Which form of covariance should we use? **A**: We use a diagonal form for the covariance matrix given the high dimension of the features (1024) and the limited number of samples in the base datasets. Even when using ImageNet, computing a full covariance matrix is undetermined. Reducing the dimensionality of the features with methods such as PCA led to poor results.

Q4: How important is the choice of α and β? **A**: Our results show that the importance of α and β varies depending on two things. The first one is the class setting: for one-class tasks, we observed that the best hyperparameters are $\alpha = 0.1$ and $\beta = 1$ which confirms the importance of the covariance matrix in this task. For the multi-class setting, we obtain different best hyperparamters depending on the task. We use the validation set on each task to select these hyperparameters. The second factor is the domain shift, we also observed that the importance of α and β vary depending on the domain gap between the base and test datasets.

6 Conclusion

In conclusion, our work takes a step toward more fully integrating text as a modality in few-shot learning, focusing on its ability to predict mean and covariance statistics for visual feature distributions. The preliminary results demonstrate improvements in robustness and generalizability across different datasets, without claiming a definitive solution to the challenges in the field. For future work, exploring higher-order moments and investigating the applicability of our approach in tasks other than classification could offer further insights into the synergistic use of text and visual data.

References

[1] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/radford21a.html`.

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[3] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[4] Sheng Shen, Chunyuan Li, Xiaowei Hu, Yujia Xie, Jianwei Yang, Pengchuan Zhang, Zhe Gan, Lijuan Wang, Lu Yuan, Ce Liu, et al. K-lite: Learning transferable visual models with external knowledge. *Advances in Neural Information Processing Systems*, 35:15558–15573, 2022.

[5] Aoxue Li, Weiran Huang, Xu Lan, Jiashi Feng, Zhenguo Li, and Liwei Wang. Boosting few-shot learning with adaptive margin loss. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2020.

[6] 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.

[7] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022.

[8] Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. Prompt learning with optimal transport for vision-language models. In *ICLR*, 2023.

[9] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7959–7971, June 2022.

[10] 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. *arXiv preprint arXiv:2110.04544*, 2021.

[11] Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

[12] Zhiqiu Lin, Samuel Yu, Zhiyi Kuang, Deepak Pathak, and Deva Ramanan. Multimodality helps unimodality: Cross-modal few-shot learning with multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19325–19337, 2023.

[13] Renrui Zhang, Xiangfei Hu, Bohao Li, Siyuan Huang, Hanqiu Deng, Yu Qiao, Peng Gao, and Hongsheng Li. Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15211–15222, 2023.

[14] Jingyi Xu and Hieu Le. Generating representative samples for few-shot classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9003–9013, 2022.

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

[16] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[17] Fengyuan Yang, Ruiping Wang, and Xilin Chen. SEGA: Semantic guided attention on visual prototype for few-shot learning. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, January 2022.

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[19] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018.

[20] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[21] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

[22] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer, 2014.

[23] 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.

[24] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[25] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[26] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[27] 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.

[28] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[29] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[30] Shell Xu Hu, Da Li, Jan Stühmer, Minyoung Kim, and Timothy M Hospedales. Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9068–9077, 2022.

[31] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[32] Yassir Bendou, Yuqing Hu, Raphael Lafargue, Giulia Lioi, Bastien Pasdeloup, Stéphane Pateux, and Vincent Gripon. Easy—ensemble augmented-shot-y-shaped learning: State-of-the-art few-shot classification with simple components. *Journal of Imaging*, 8(7):179, 2022.

A Supplementary Material

A.1 Data collection

We use instruct-GPT [16] to generate a visual description of each class. We use different prompts depending on the dataset:

Dataset	Prompt Used
iNaturalist	How can you identify a {class}
Pets	How can you identify a {class}?
SUN397	How can you identify a {class}?
Eurosat	How can you identify a satellite image of a {class}?
DTD	How can you identify a {class} texture?
Flowers	How can you identify a {class} flower?
Caltech101	How can you identify a {class}?
Food101	How can you identify a {class}?
Stanford Cars	How can you identify a {class} car?
UCF101	How can you identify a {class}?
Imagenet	How can you identify a {class}?

Table 4: Datasets and Their Corresponding Prompts

A.2 Normalization of the input

In this section, we detail how the normalization is performed. On the base classes, we compute the mean of the input texts and targets i.e. class means and covariances respectively. After that, we apply a Z-score normalization as follows:

$$\forall x \in \mathbb{R}^{Nd} : Z_{i,j} = \frac{x_{ij} - \hat{\mu}_j}{\hat{\sigma}_j} \tag{3}$$

where i is the i-th datapoint and j is the j-th feature.