Supplementary Material

1. Proof of the proposed approximation

The details of how we obtain the model presented in equation (5) in the main paper can be found below:

$$
\begin{aligned}
w_s &= p(\mathbf{x} \in \mathcal{D}_s) \\
&= p(s' = s|\mathbf{x}) \\
&= \frac{p(\mathbf{x}|s) \cdot p(s)}{p(\mathbf{x})} \text{ (Bayes theorem)} \\
&= \frac{p(\mathbf{x}|s) \cdot p(s)}{\sum_j p(\mathbf{x}|j) \cdot p(j)} \text{ (Marginalization)} \\
&= \frac{p(\mathbf{x}|s)}{\sum_j p(\mathbf{x}|j)} \ (\mathcal{H}_1) \\
&= \frac{p(\Delta_s(\mathbf{x})|s)}{\sum_j p(\Delta_j(\mathbf{x})|j)} \ (\mathcal{H}_2) \\
&= \frac{\mathcal{N}(\Delta_s(\mathbf{x}); \mu_s^{k^*}, \sigma_s^{k^*})}{\sum_j \mathcal{N}(\Delta_j(\mathbf{x}); \mu_j^{t^*}, \sigma_j^{t^*})},
\end{aligned}
\tag{1}
$$

We have to make three assumptions or hypothesis to derive this model:

- $\mathcal{H}_1$: Each domain is of equal importance in our scenario, i.e. if we consider the probability of the sample belonging to a certain domain uniform when we have no a priori on the sample.
- $\mathcal{H}_2$: $p(\mathbf{x}|s) \approx p(\Delta_s(\mathbf{x})|s)$, i.e. the distribution of $f_\theta(\mathbf{x}_{\text{tok}})$ with $x_{\text{tok}} \in \mathcal{D}_s$ is isotropic.
- $\mathcal{H}_3$: $\Delta_s(\boldsymbol{x})|s \sim \mathcal{N}(\cdot; \mu_s^{k^*}, \sigma_s^{k^*})$, i.e. $\boldsymbol{x})|s$ follows a Gaussian of mean $\mu_s^{k^*}$ and standard deviation $\sigma_s^{k^*}$.

$\mathcal{H}_1$ is reasonable in practice as test sample can come from any domain with equal probability. $\mathcal{H}_2$ and $\mathcal{H}_3$ are made to simplify the model, make it easy to store in memory and to compute. These hypothesis transform the mixture weights model into a Gaussian Mixture Model on the distances to the prototypes (L2-GMM). Please note that in our case the ensembling with the Mahanalobis distance is equivalent to the well known classical GMM using directly the features and the prototypes to derive $p(\mathbf{x} \in \mathcal{D}_s)$.

We empirically observe in the ablation study (Table (4) in the main paper) that the usage of this Gaussian Mixture Model on the distances to the prototypes yields superior performance compared to a GMM using directly the features and the prototypes. We suspect that these approximations are efficient because they reduce the coordinate-wise noise in the standard deviations inherent to the Mahanalobis distance. Gaussian seems like a good approximation of $\Delta_s(\boldsymbol{x})|s$, even though the approximation using other distributions could be investigated in the future, such as the Weibull Distribution or the Generalized Pareto Distribution.

2. Algorithm

The detailed algorithm of the proposed MoP-CLIP approach is shown in Algorithm 1. $\mathbf{x}$ denotes the samples to be classified, f_θ and f_ϕ the visual and text encoder of the network and $\mathcal{P}^V$, $\mathcal{P}^T$ the sets of visual of text prompts and $\mathcal{E}$ the domains prototypes learned during training. $\mathcal{G} = \{(\mu_s^k; \sigma_s^k), s = 1..N, k = 1..K\}$ denotes the parameters of the Gaussian distributions learned for the different domains s and classes k.

Algorithm 1 Inference procedure for the proposed method

Input: $\mathbf{x}$; f_θ; f_ϕ; $\mathcal{P}^V$; $\mathcal{P}^T$; $\mathcal{E}$; $\mathcal{G}$;
Init $E \in O^{K \times N}$
Compute image features: $f_x \leftarrow f_\theta(\mathbf{x}_{tok})$
Compute matrix $D : D_{i,j} \leftarrow ||f_x - \boldsymbol{m}_j^i||_2$
Compute matrix $D' : D'_j \leftarrow \min_i D_{i,j}$
if $F(\Delta_{s^*}(\mathbf{x})) \leq q$ ($\mathbf{x}$ is In-Domain) **then**
  $W_{s^*} = 1, \forall s \neq s^*, W_s = 0.$
  Compute prediction using the best prompt:
  for $k = 1, 2, ..., K$ **do**
    $\mathbf{x}_{pro} \leftarrow [\mathbf{x}_{tok}, \mathbf{p}_{s^*}^v, x_{cls}]$
    $t_j \leftarrow [\mathbf{p}_{s^*}^t, c_j]$
    $E_{k,s^*} \leftarrow \frac{\exp(cos(f_\theta(\mathbf{x}_{pro}), f_\phi(t_k)))}{\sum_{i=1}^{C} \exp(cos(f_\theta(\mathbf{x}_{pro}), f_\phi(t_i)))}$
  end for
else
  Compute W using equation (5), D' and $\{(\mu_s^{k^*}, \sigma_s^{k^*})\}_{s=1}^N$.
  Compute predictions using the different prompts:
  for $s = 1, 2, ..., N$ **do**
    for $k = 1, 2, ..., K$ **do**
      $\mathbf{x}_{pro} \leftarrow [\mathbf{x}_{tok}, \mathbf{p}_s^v, x_{cls}]$
      $t_j \leftarrow [\mathbf{p}_s^t, c_j]$
      $E_{k,s} \leftarrow \frac{\exp(cos(f_\theta(\mathbf{x}_{pro}), f_\phi(t_k)))}{\sum_{i=1}^{C} \exp(cos(f_\theta(\mathbf{x}_{pro}), f_\phi(t_i)))}$
    end for
  end for
end if
$P \leftarrow E \cdot W^T$ Return P the soft classification vector

3. Additional results

Table 1 emphasizes that S-Prompts performances degrade when evaluation is done on unseen domains, and shows that the proposed MoP-CLIP seems to generalize better, mitigating the performance degradation under domain distributions. In particular, the left-side section reports the results of S-Prompts trained separately on the different domains (*x-axis*) and evaluated in each of the domains (*y-axis*). For example, 67.41 denotes the accuracy of the model trained solely on Infograph domain and tested on the Clipart domain. We use blue to denote the performance of

Table 1. **Empirical motivation of resorting to the prediction ensembling scheme for OOD situations.** Classification accuracy across DomainNet domains using different specialized prompts, for both single and ensembling predictions. The results blue denote the accuracy with the in-domain prompts, whereas results in magenta denote the accuracy using the best out-of-domain prompts (prompts from all domains except the current one). Furthermore, results in bold (*last column*) denote the highest accuracy amongst out-of-domain methods. For 5 out of 6 domain sets, the proposed prediction ensembling method yields higher accuracy than the best out-of-domain prompt. This suggests that the ensembling technique is overall relevant when test examples are from a novel domain (i.e. unseen during the training).

	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	S-Prompts (ID)	S-Prompts (OOD)	Pred. Ens. (OOD)
Clipart	80.14	67.41	64.77	38.90	69.49	69.02	78.57	69,31	**73.48** (+4.01)
Infograph	44.59	60.65	43.24	15.36	48.93	36.08	58.72	46.50	**50.40** (+1.47)
Painting	59.56	61.88	78.00	24.97	64.43	57.32	74.76	61.88	**67.93** (+3.50)
Quickdraw	16.80	13.11	8.30	46.65	13.58	**17.29**	46.59	16.79	16.78 (−0.51)
Real	78.35	79.38	75.83	45.44	87.94	71.79	85.19	77.38	**83.48** (+4.10)
Sketch	61.51	59.18	55.22	30.43	61.59	72.97	69.76	58.87	**66.31** (+4.72)

in-distribution samples (when train and test data are drawn from the same distribution), which can be considered as an upper bound, as there is no distributional drift between samples. Then, both results in black and magenta highlight the results for each tested domain, assuming that the tested domain remains unknown and all training samples come from the same domain (specified in each column). Note that across each test domain we highlight the results from the best model in magenta. If we look at the results obtained by S-Prompts under ID and OOD conditions (*S-Prompts (ID)* and *S-Prompts (OOD)* columns), we can observe that: *i)* its performance deteriorates under domain shift and *ii)*, the selection criterion of S-Prompts is not always optimal. On the other hand, the proposed approach (*last column*) substantially outperforms S-Prompts in five out of six domains, as well as the best out-of-distribution model (in magenta).

4. Analysis of memory-space complexity

One of the main benefits of the proposed approach compared to memory replay strategies is its much lighter storage requirements. Indeed, memory replay methods typically store 50 images as exemplars per class and per domain, which amounts for 15M float numbers per domain in the case of CDDB (2 classes per domain and image dimensions of $3 \times 224 \times 224$). In contrast, MoP-CLIP stores one prototype per class per domain, along with one distance mean and one distance standard deviation. We additionally store a set of (10×768 and 16×768) visual and text prompts per domain, which, in total, amounts for 0.02M float numbers per domain (in CDDB).

5. Impact of the prompt length

We used $L^v = 10$ in all of our experiments to compare fairly with [39]. However, we now study the impact of L^v on the Average Accuracy on the CDDB-Hard dataset, whose results are depicted in Fig. 1. These results emphasize that *i)* the performance of the proposed MoP-CLIP is more stable than S-Prompts [39] with respect to L^v, and *ii)* that our method systematically yields higher performance on the Unseen Domains. We can therefore state that our proposed MoP-CLIP is more robust in real world scenarios.

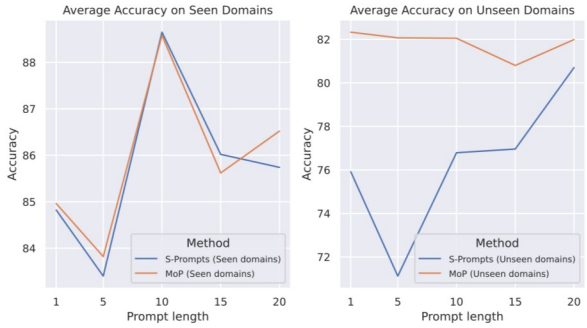

Figure 1. Impact of the prompt length L^v (Sec. 3.2) on the Average Accuracy, evaluated on the ID and OOD domains of CDDB-Hard.