

FewVS: A Vision-Semantics Integration Framework for Few-Shot Image Classification (Supplementary Material)

Anonymous Author(s)

A IMPLEMENTATION DETAILS

A.1 Details of the Optimal Transport-based Training

Here, we provide a pseudo-code for FewVS training loop in PyTorch style, including the detail of Sinkhorn-Knopp algorithm [1]:

```
# E: entity embeddings (DxL)
# Enc_f: few-shot encoder
# Enc_v: CLIP's vision encoder
# h: linear projection layer
# temp: temperature

for x in loader: # load a batch x with B samples
    z_t = Enc_v(x) # target image features: BxD
    z_s = h(Enc_f(x)) # source image projections: BxD

    scores_t = mm(z_t, E) # embedding scores: BxL
    scores_s = mm(z_s, E) # embedding scores: BxL

    # compute assignments
    with torch.no_grad():
        q_t = sinkhorn(scores_t)
        q_s = sinkhorn(scores_s)

    # convert scores to probabilities
    p_t = Softmax(scores_t / temp)
    p_s = Softmax(scores_s / temp)

    # swap prediction problem
    loss = - 0.5 * mean(q_t * log(p_s) + q_s * log(p_t))

    # SGD update: linear projection layer
    loss.backward()
    update(h.params)

# Sinkhorn-Knopp
def sinkhorn(scores, eps=0.05, niters=3):
    Q = exp(scores / eps).T
    Q /= sum(Q)
    K, B = Q.shape
    u, r, c = zeros(K), ones(K) / K, ones(B) / B
    for _ in range(niters):
        u = sum(Q, dim=1)
        Q *= (r / u).unsqueeze(1)
        Q *= (c / sum(Q, dim=0)).unsqueeze(0)
    return (Q / sum(Q, dim=0, keepdim=True)).T
```

A.2 Prompting the Language Model

We mine fine-grained semantic attributes from the language model. Specifically, we utilize the "gpt-3.5-turbo-instruct" model, setting the temperature to 0 and limiting the token count to 100. In line with OpenAI's API guidelines, we format our query and expected response using the structure "Q:A:". Following [3], we append a trailing hyphen to facilitate the creation of a bulleted list output. This formatting simplifies the extraction of semantic attributes by merely removing the hyphens. Additionally, we include examples

of the formatted responses to enhance the model's output through in-context learning [2]. The complete input text is as follows:

```
Q: What are visual details for distinguishing a house
    finch in a photo?
A: There are several useful visual details of a house
    finch in a photo:
- small bird
- red, brown, or grey
- conical beak
- notched tail
- a short, forked tail
- a brown or grey back
- chest with streaks or spots
- white underbelly
- sometimes a red or orange head and chest in males
- typically found perching or feeding in urban or
  suburban area

Q: What are visual details for distinguishing a
    jellyfish in a photo?
A: There are several useful visual details of a
    jellyfish in a photo:
- umbrella-like shape
- translucent or transparent body
- usually lacks solid structure
- tentacles with stinging cells underneath
- sometimes bright colors
- movement by pulsating motion
- spherical shape
- white with colored panels
- typically 18 panels with 6 equal sections of three
  panels
- texture for grip

Q: What are visual details for distinguishing a [
    class name] in a photo?
A: There are several useful visual details of a [
    class name] in a photo:
```

B MORE ABLATION STUDIES

B.1 Impact of the Weight Factor

The weight factor α in Eq.(9) determines the balance between visual and semantic information. We analyze the effect of α on classification accuracy by setting it to different values. Note that $\alpha = 0$ implies that the few-shot task degenerates to a zero-shot task, relying solely on semantic information for classification. As demonstrated in Figure 1, when a larger number of support samples is available (*i.e.*, 5-shot task), representative prototypes can be extracted, and a larger α should be set. Conversely, when fewer support samples are available (*i.e.*, 1-shot task), classification relies mainly on semantic information, necessitating a smaller α .

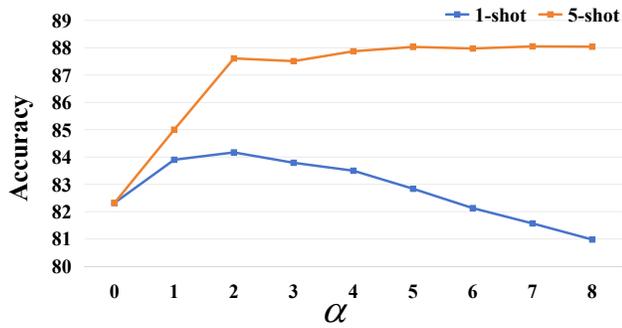


Figure 1: Impact of α with FewVS-Res on *tieredImageNet*.

B.2 Impact of the Number of Iterations in Online Vision-Semantics Integration

In FewVS, we optimize ω in Eq.(11) by iteratively using gradient descent with the support set as supervision. Here, we explore the impact of the number of iterations on classification accuracy. As shown in Figure 2, our online vision-semantics integration module consistently improves FSIC by adaptively integrating fine-grained semantic attributes and visual information. However, setting a high number of iterations has a relatively negative effect on the 1-shot task. This is because excessive gradient descents can cause the elements within ω^* to assume large values, significantly adjusting the contribution of each semantic attribute to classification. However, given the limited support samples in the 1-shot task, it is difficult to

accurately assess the positive or negative impacts of each semantic attribute on the current task. This leads to an incorrect adjustment of the contributions of semantic attributes.

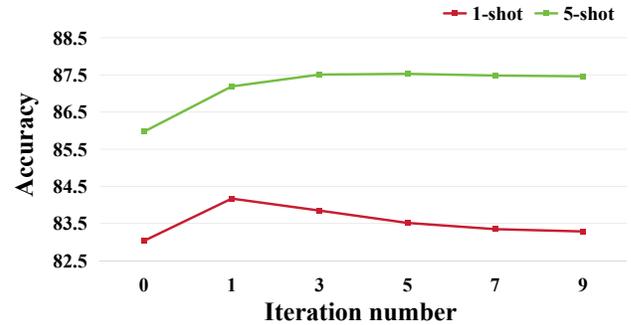


Figure 2: Impact of the number of iterations in online vision-semantics integration with FewVS-Res on *tieredImageNet*.

REFERENCES

- [1] Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems* 26 (2013).
- [2] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234* (2022).
- [3] Sachit Menon and Carl Vondrick. 2022. Visual classification via description from large language models. *arXiv preprint arXiv:2210.07183* (2022).