PromptDet: Towards Open-vocabulary Detection using Uncurated Images

Chengjian Feng¹, Yujie Zhong¹, Zequn Jie¹, Xiangxiang Chu¹
Haibing Ren¹, Xiaolin Wei¹, Weidi Xie², and Lin Ma¹

¹Meituan Inc. ²Shanghai Jiao Tong University
Detection

• Localization
• Classification
Open-vocabulary

- **Open-set**
  - Train on A, test on B

- **Multimodality**
  - Visual, image/video
  - Language, text

- Prediction related with input text prompt
CLIP: Connecting Text and Images

(1) Contrastive pre-training

(2) Create dataset classifier from label text

(3) Use for zero-shot prediction
CLIP: Connecting Text and Images

(1) Contrastive pre-training

Text Encoder

Image Encoder

Transformer

[SOS]  a  photo  of  dog  [EOS]
CLIP: Connecting Text and Images

(1) Contrastive pre-training

Text Encoder

Image Encoder

Transformer

H * W
CLIP: Connecting Text and Images

(1) Contrastive pre-training

```python
# image_encoder - ResNet or Vision Transformer
text_encoder - CBOW or Text Transformer
I[n, h, w, c] - minibatch of aligned images
T[n, 1] - minibatch of aligned texts
W_i[d_i, d_e] - learned proj of image to embed
W_t[d_t, d_e] - learned proj of text to embed
t - learned temperature parameter

# extract feature representations of each modality
I_f = image_encoder(I) #[n, d_i]
T_f = text_encoder(T) #[n, d_t]

# joint multimodal embedding [n, d_e]
I_e = l2_normalize(np.dot(I_f, W_i), axis=1)
T_e = l2_normalize(np.dot(T_f, W_t), axis=1)

# scaled pairwise cosine similarities [n, n]
logits = np.dot(I_e, T_e.T) * np.exp(t)

# symmetric loss function
labels = np.arange(n)
loss_i = cross_entropy_loss(logits, labels, axis=0)
loss_t = cross_entropy_loss(logits, labels, axis=1)
loss = (loss_i + loss_t)/2
```
CLIP: Connecting Text and Images

• Dataset
  • 400 million pairs (image, text)
  • Collected from Internet

• Training
  • 32 epochs
  • minibatch size of 32,768
  • 18 days on 592 V100 GPUs (RN50x64)
  • 12 days on 256 V100 GPUs (ViT-L/14)
PromptDet: Towards Open-vocabulary Detection using Uncurated Images

Chengjian Feng\textsuperscript{1}, Yujie Zhong\textsuperscript{1}, Zequn Jie\textsuperscript{1}, Xiangxiang Chu\textsuperscript{1}
Haibing Ren\textsuperscript{1}, Xiaolin Wei\textsuperscript{1}, Weidi Xie\textsuperscript{2, \$}, and Lin Ma\textsuperscript{1}

\textsuperscript{1}Meituan Inc. \hspace{0.5cm} \textsuperscript{2}Shanghai Jiao Tong University
Naive Detector

- Class-agnostic RPN
- Classify with distance
Naive Detector

- Class-agnostic RPN
- Classify with distance

\[
c_{\text{almond}} = \phi_{\text{text}}(g(\text{“this is a photo of [almond]”}))
\]

\[
c_{\text{dog}} = \phi_{\text{text}}(g(\text{“this is a photo of [dog]”}))
\]

\[
p_{\text{almond}} = \frac{\exp(<v, c_{\text{almond}}>/\epsilon)}{\exp(<v, c_{\text{almond}}>/\epsilon) + \exp(<v, c_{\text{dog}}>/\epsilon)}
\]
Limitation

• {Class name} → category embedding, suboptimal
  • Lexical ambiguity

• Domain gap on the visual representation
  • CLIP, scene-centric
  • RPN, object-centric

• Base categories, less diverse
  • Insufficient to guarantee the generalization
Regional Prompt Learning RPL

• Learnable vectors
  • Shared for all categories

• Description

\[ c_{\text{almond}} = \phi_{\text{text}}([p_1, \ldots, p_j, g(\text{category}), p_{j+1} \ldots, p_{j+h}, g(\text{description})]) \]

{category: “almond”, description: “oval-shaped edible seed of the almond tree”}
Regional Prompt Learning RPL

- Off-line manner
- Visual
  - Crops from LVIS
  - Base categories
Regional Prompt Learning RPL

Table 2: Comparison on manually designed and learned prompt. Here, we only use two learnable prompt vectors in PRL, i.e. $[1 + 1]$ refers to using one vector for prefix, and one vector for suffix.

<table>
<thead>
<tr>
<th>Prompt</th>
<th>AP_{novel}</th>
<th>AP_c</th>
<th>AP_f</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>“a photo of [category]”</td>
<td>manual</td>
<td>7.4</td>
<td>17.2</td>
<td>26.1</td>
</tr>
<tr>
<td>“a photo of [category], which is [description]”</td>
<td>manual</td>
<td>9.0</td>
<td>18.6</td>
<td>26.5</td>
</tr>
<tr>
<td>regional prompt learning</td>
<td>[1+1]</td>
<td>11.1</td>
<td>18.8</td>
<td>26.6</td>
</tr>
</tbody>
</table>
Self-training

- Sourcing candidate images
  - LAION-400M, initial corpus
  - N novel category prompt
  - Keep images with highest similarity
  - w/o GT box

- Alternate RPL and sourcing
Self-training

- Alternate RPL and sourcing

Table 3: Effectiveness of self-training with different prompts. 1-iter, 2-iter and 3-iter denote that Stage-I (*i.e.* RPL) and Stage-II (*i.e.* image sourcing) are performed for one, two or three iterations, respectively.

<table>
<thead>
<tr>
<th>Prompt method</th>
<th>Self-training</th>
<th>AP\textsubscript{novel}</th>
<th>AP\textsubscript{c}</th>
<th>AP\textsubscript{f}</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>“a photo of [category], which is [description]”</td>
<td>✓</td>
<td>9.0</td>
<td>18.6</td>
<td>26.5</td>
<td>20.1</td>
</tr>
<tr>
<td>Regional prompt learning</td>
<td></td>
<td>11.1</td>
<td>18.8</td>
<td>26.6</td>
<td>20.3</td>
</tr>
<tr>
<td>PromptDet (1-iter)</td>
<td>✓</td>
<td>15.3</td>
<td>17.7</td>
<td>25.8</td>
<td>20.4</td>
</tr>
<tr>
<td>PromptDet (2-iter)</td>
<td>✓</td>
<td>19.0</td>
<td>18.5</td>
<td>25.8</td>
<td>21.4</td>
</tr>
<tr>
<td>PromptDet (3-iter)</td>
<td>✓</td>
<td>19.3</td>
<td>18.3</td>
<td>25.8</td>
<td>21.4</td>
</tr>
</tbody>
</table>
Self-training

- Bounding box generation
  - Sourced images, object-centric
  - Top-K proposals, objectness
  - Maximal classification score
  - Re-training
Self-training

Fig. 4: Visualisation of the generated pseudo ground truth for the sourced images.
Self-training

Table 4: **Left**: the comparison on different box generation methods. **Right**: the effect on increasing the sourced candidate images.

<table>
<thead>
<tr>
<th>Method</th>
<th>$\text{AP}_{\text{novel}}$</th>
<th>$\text{AP}_c$</th>
<th>$\text{AP}_f$</th>
<th>$\text{AP}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o self-training</td>
<td>10.4</td>
<td>19.5</td>
<td>26.6</td>
<td>20.6</td>
</tr>
<tr>
<td>image</td>
<td>9.9</td>
<td>18.8</td>
<td>26.0</td>
<td>20.1</td>
</tr>
<tr>
<td>max-size</td>
<td>9.5</td>
<td>18.8</td>
<td>26.1</td>
<td>20.1</td>
</tr>
<tr>
<td>max-obj.-score</td>
<td>11.3</td>
<td>18.7</td>
<td>26.0</td>
<td>20.3</td>
</tr>
<tr>
<td>max-pred.-score (ours)</td>
<td>19.0</td>
<td>18.5</td>
<td>25.8</td>
<td>21.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#Web images</th>
<th>$\text{AP}_{\text{novel}}$</th>
<th>$\text{AP}_c$</th>
<th>$\text{AP}_f$</th>
<th>$\text{AP}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>10.4</td>
<td>19.5</td>
<td>26.6</td>
<td>20.6</td>
</tr>
<tr>
<td>50</td>
<td>14.6</td>
<td>19.3</td>
<td>26.2</td>
<td>21.2</td>
</tr>
<tr>
<td>100</td>
<td>15.8</td>
<td>19.3</td>
<td>26.2</td>
<td>21.4</td>
</tr>
<tr>
<td>200</td>
<td>17.4</td>
<td>19.1</td>
<td>26.0</td>
<td>21.5</td>
</tr>
<tr>
<td>300</td>
<td>19.0</td>
<td>18.5</td>
<td>25.8</td>
<td>21.4</td>
</tr>
</tbody>
</table>
Dataset

Table 1: A summary of dataset statistics. The numbers in bracket refer to the number of base and novel categories.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th>Eval</th>
<th>Definition</th>
<th>#Images</th>
<th>#Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>LVIS</td>
<td>–</td>
<td>–</td>
<td>original LVIS dataset</td>
<td>0.1M</td>
<td>1203</td>
</tr>
<tr>
<td>LAION-400M</td>
<td>–</td>
<td>–</td>
<td>image-text pairs filtered by CLIP</td>
<td>400M</td>
<td>unlabeled</td>
</tr>
<tr>
<td>LVIS-base</td>
<td>✓</td>
<td>✗</td>
<td>common and frequent categories</td>
<td>0.1M</td>
<td>866</td>
</tr>
<tr>
<td>LAION-novel</td>
<td>✓</td>
<td>✗</td>
<td>image subset of novel categories</td>
<td>0.1M</td>
<td>337 (noisy)</td>
</tr>
<tr>
<td>LVIS minival</td>
<td>✗</td>
<td>✓</td>
<td>standard LVIS validation set</td>
<td>20K</td>
<td>1203 (866+337)</td>
</tr>
</tbody>
</table>
Table 6: Detection results on the LVIS v1.0 validation set. Both Detic and our proposed approach have exploited the external images. However, in Detic, the images are manually annotated and thus indicated by ‘*’. Notably, PromptDet does not require a knowledge distillation from the CLIP visual encoder at the detector training, which is shown to prominently boost the performance but significantly increase the training costs.

<table>
<thead>
<tr>
<th>Method</th>
<th>Epochs</th>
<th>Scale Jitter</th>
<th>Input Size</th>
<th>#External</th>
<th>AP_{novel}</th>
<th>AP_{c}</th>
<th>AP_{f}</th>
<th>AP</th>
</tr>
</thead>
<tbody>
<tr>
<td>ViLD-text [11]</td>
<td>384</td>
<td>100~2048</td>
<td>1024×1024</td>
<td>0</td>
<td>10.1</td>
<td>23.9</td>
<td>32.5</td>
<td>24.9</td>
</tr>
<tr>
<td>ViLD [11]</td>
<td>384</td>
<td>100~2048</td>
<td>1024×1024</td>
<td>0</td>
<td>16.1</td>
<td>20.0</td>
<td>28.3</td>
<td>22.5</td>
</tr>
<tr>
<td>ViLD-ens. [11]</td>
<td>384</td>
<td>100~2048</td>
<td>1024×1024</td>
<td>0</td>
<td>16.6</td>
<td>24.6</td>
<td>30.3</td>
<td>25.5</td>
</tr>
<tr>
<td>Detic [36]</td>
<td>384</td>
<td>100~2048</td>
<td>1024×1024</td>
<td>1.2M*</td>
<td>17.8</td>
<td>26.3</td>
<td>31.6</td>
<td>26.8</td>
</tr>
<tr>
<td>PromptDet</td>
<td>12</td>
<td>640~800</td>
<td>800×800</td>
<td>0.1M</td>
<td>19.0</td>
<td>18.5</td>
<td>25.8</td>
<td>21.4</td>
</tr>
<tr>
<td>PromptDet</td>
<td>72</td>
<td>100~1280</td>
<td>800×800</td>
<td>0.1M</td>
<td><strong>21.4</strong></td>
<td>23.3</td>
<td>29.3</td>
<td>25.3</td>
</tr>
</tbody>
</table>