A. Appendix / Supplemental Material

A.1. Pseudocode

In this section, we provide pseudocodes to illustrate the workflow of PSA. Algorithm 1 presents the initialization process, detailing how discrete prototypes are identified from paired image–text data, while Algorithm 2 demonstrates the query–response mechanism generating approximate textual semantics purely from image embeddings.

Algorithm 1 PSA Initialization

Define:
  - A set of K paired samples $\{(I_i, T_i)\}_{i=1}^{K}$;
  - image-only samples $\{I_j\}$;
  - pretrained encoders f_{enc}^{I} and f_{enc}^{T};
  - cross-attention module (Language-Guided U-Net)
  - token selection threshold τ;
  - number of semantic clusters N (HDBSCAN);
  - number of sub-clusters M (K-means).
Return: Prototype space $\mathcal{S} = (\mathcal{S}^Q, \mathcal{S}^R)$
Step 1: Encode Paired Samples
for $i = 1$ to K **do**
  $e_i^I \leftarrow f_{\text{enc}}^{I}(I_i)$
  $e_i^T \leftarrow f_{\text{enc}}^{T}(T_i)$
end for
Step 2: Extract Segmentation-Relevant Tokens
for $i = 1$ to K **do**
  Compute cross-attention scores α_j for each token t_j in T_i
  $T_i^{\text{selected}} \leftarrow \{t_j \mid \alpha_j > \tau\}$
  $e_i^{\text{sem}} \leftarrow f_{\text{enc}}^{T}(T_i^{\text{selected}})$
end for
Step 3: Cluster Textual Semantics (HDBSCAN)
Perform HDBSCAN on $\{e_i^{\text{sem}}\}$ to form N clusters $\{\mathcal{C}_1, \ldots, \mathcal{C}_N\}$
Step 4: Form Image Sub-Clusters (K-means)
$\mathcal{S}^Q \leftarrow \emptyset, \quad \mathcal{S}^R \leftarrow \emptyset$
for $i = 1$ to N **do**
  Extract embeddings $\{(e_j^I, e_j^T) \mid j \in \mathcal{C}_i\}$
  Run K-means with M sub-clusters: $\mathcal{C}_{i1}, \ldots, \mathcal{C}_{iM}$
  for $j = 1$ to M **do**
    Identify representative $c_{ij} = (e_k^I, e_k^T)$ closest to sub-cluster centroid
    $q_{ij} \leftarrow e_k^I$ (query prototype)
    $r_{ij} \leftarrow e_k^T$ (response prototype)
    $\mathcal{S}^Q \leftarrow \mathcal{S}^Q \cup \{q_{ij}\}, \quad \mathcal{S}^R \leftarrow \mathcal{S}^R \cup \{r_{ij}\}$
  end for
end for
Step 5: Output Prototype Space
$\mathcal{S} \leftarrow (\mathcal{S}^Q, \mathcal{S}^R)$
return $\mathcal{S}$

Algorithm 2 PSA Query and Response

Require: Prototype space $\mathcal{S} = (\mathcal{S}^Q, \mathcal{S}^R)$; pretrained image encoder f_{enc}^{I}; Language-Guided U-Net f_{seg}; query image I^*; top-k integer k.
Ensure: Approximated textual feature r^* for guiding segmentation
Step 1: Encode the Query Image
$q^* \leftarrow f_{\text{enc}}^{I}(I^*)$
Step 2: Compute Similarity Scores
for all q_{ij} in $\mathcal{S}^Q$ **do**
  $s_{ij} \leftarrow \text{cosine\_similarity}(q^*, q_{ij})$
end for
Step 3: Select Top-k Queries
$Q^* \leftarrow \arg\text{top}_k(\{s_{ij}\})$
Step 4: Retrieve Corresponding Responses
$R^* \leftarrow \{r_{ij} \mid q_{ij} \in Q^*\}$
Step 5: Aggregate Responses (Weighted Sum)
$r^* \leftarrow \sum_{(q_{ij}, r_{ij}) \in Q^* \times R^*} w_{ij}\, r_{ij}$
where $w_{ij} = \frac{\exp(s_{ij})}{\sum_{q_{i'j'} \in Q^*} \exp(s_{i'j'})}$
return r^*

A.2. Implementation Details

Following the previous design of language-guided segmentation networks [**?** **?**], we adopt a U-Net backbone with feature fusion at the decoder stage. The image is resized into 224×224, and textual reports are tokenized, truncated, and padded to a fixed length of 256 tokens. To construct the prototype space we set the number of surrogate labels to 6, with each label containing 64 prototypes. During inference, the PSA module retrieves the top 10 prototype candidates per query for semantic approximation. We use the AdamW optimizer with an initial learning rate of 10^{-4}, which is scheduled to decay using cosine annealing.

A.3. Limitations and Future Works

In this work, we focused on demonstrating the core idea of ProLearn in single-label 2D segmentation. Future directions involve exploring multi-label and volumetric data, broader imaging modalities, and extending to more language-guided vision tasks.

A.4. Visualization

To further demonstrate the effectiveness of our proposed ProLearn, we provide additional visual comparisons of segmentation results in the next page. Specifically, we show the performance of LViT, GuideSeg, SGSeg, and our ProLearn on the QaTa-COV19 and MosMedData+ dataset under different text availability (1%, 5%, 10%, 25%, and 50%).

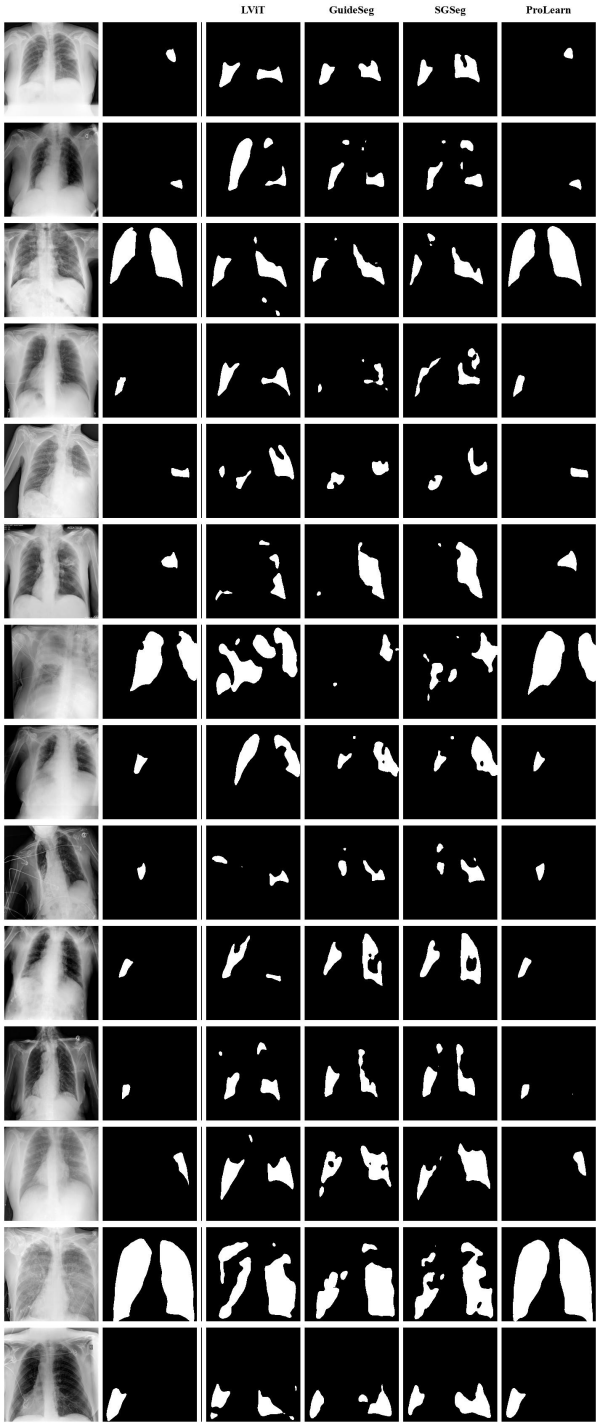

Figure A1. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on QaTa-COV19 under 1% text availability.

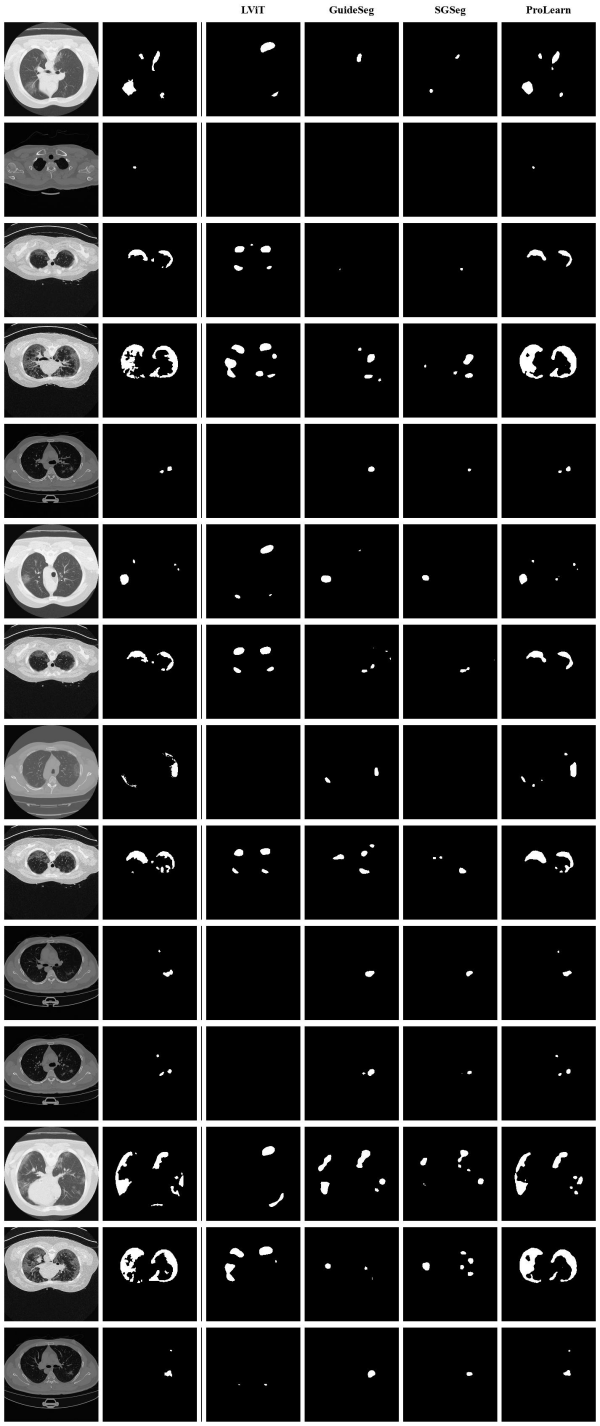

Figure A2. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on MosMedData+ dataset under 1% text availability.

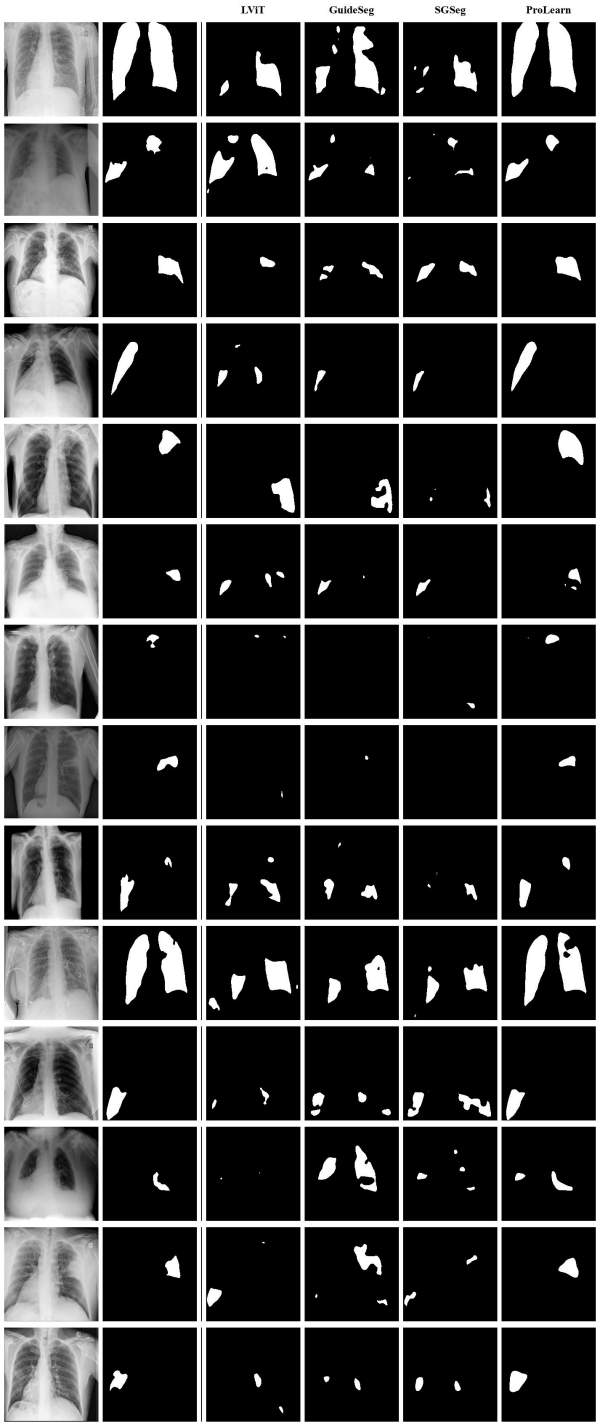

Figure A3. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on QaTa-COV19 dataset under 5% text availability.

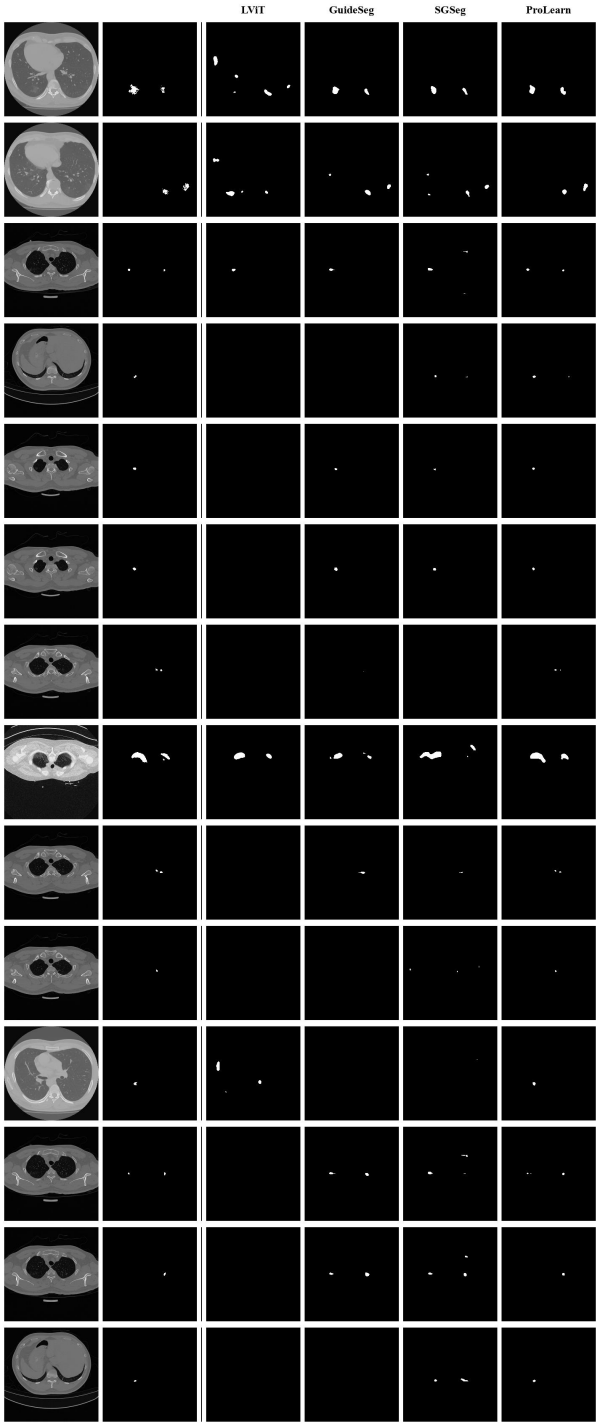

Figure A4. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on MosMedData+ dataset under 5% text availability.

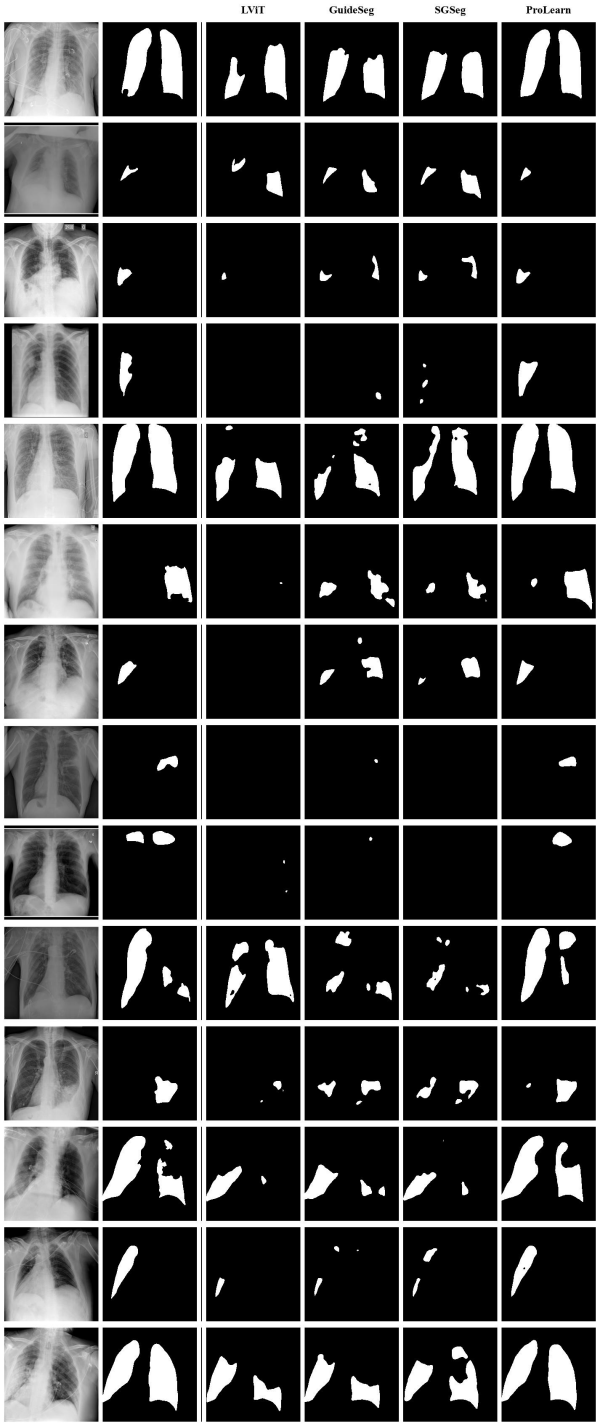

Figure A5. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on QaTa-COV19 dataset under 10% text availability.

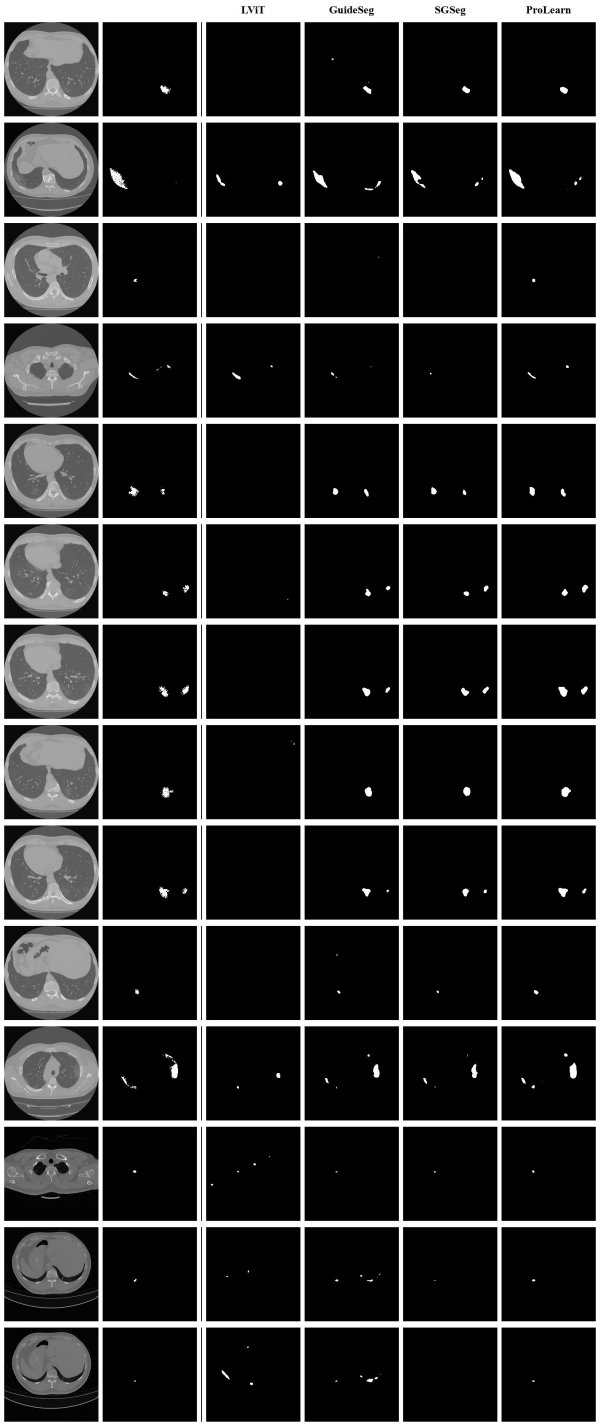

Figure A6. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on MosMedData+ dataset under 10% text availability.

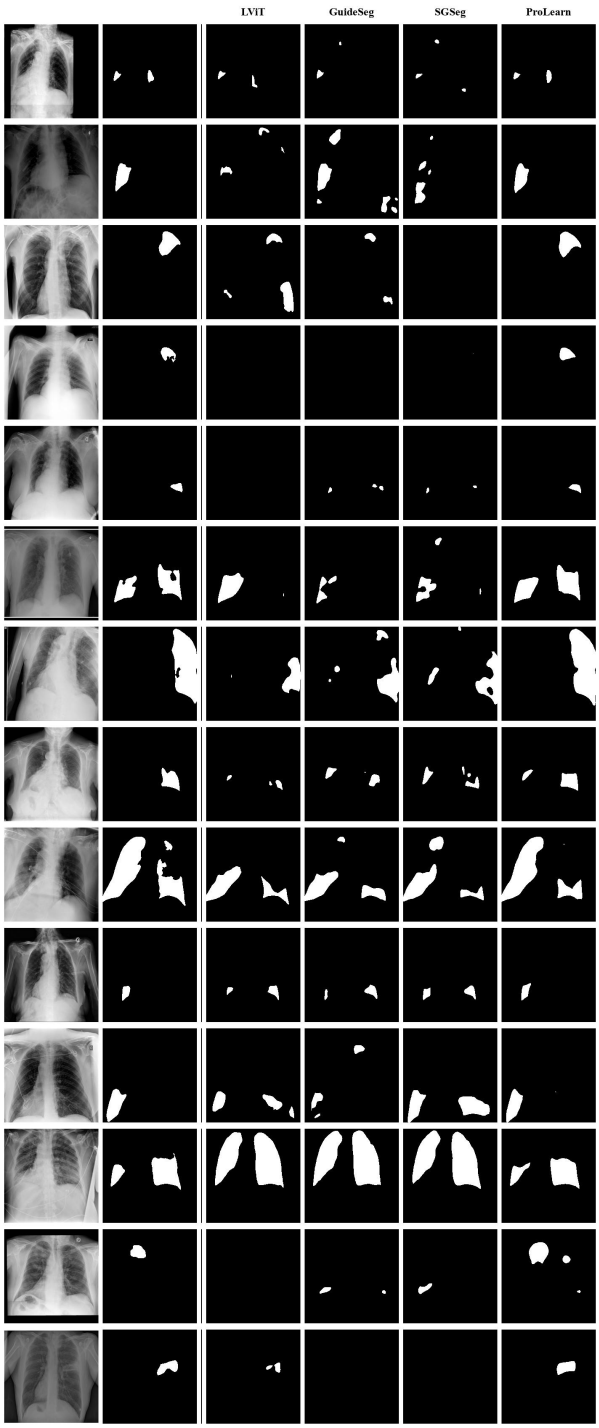

Figure A7. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on QaTa-COV19 dataset under 25% text availability.

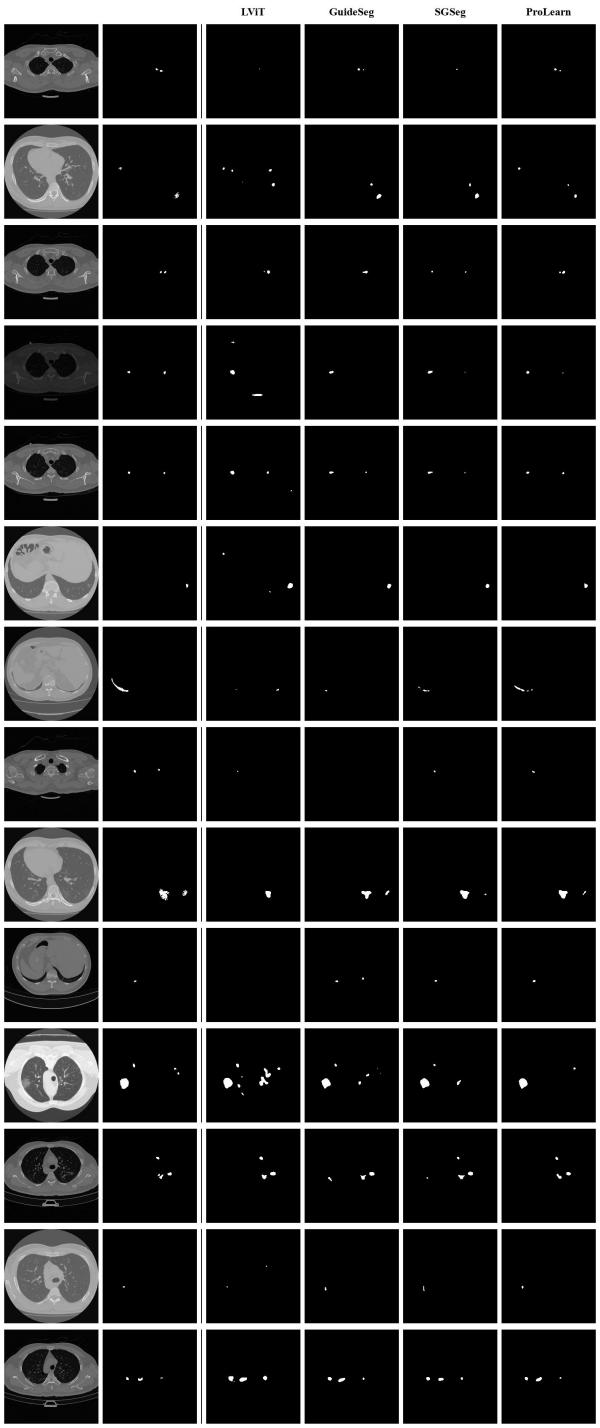

Figure A8. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on MosMedData+ dataset under 25% text availability.

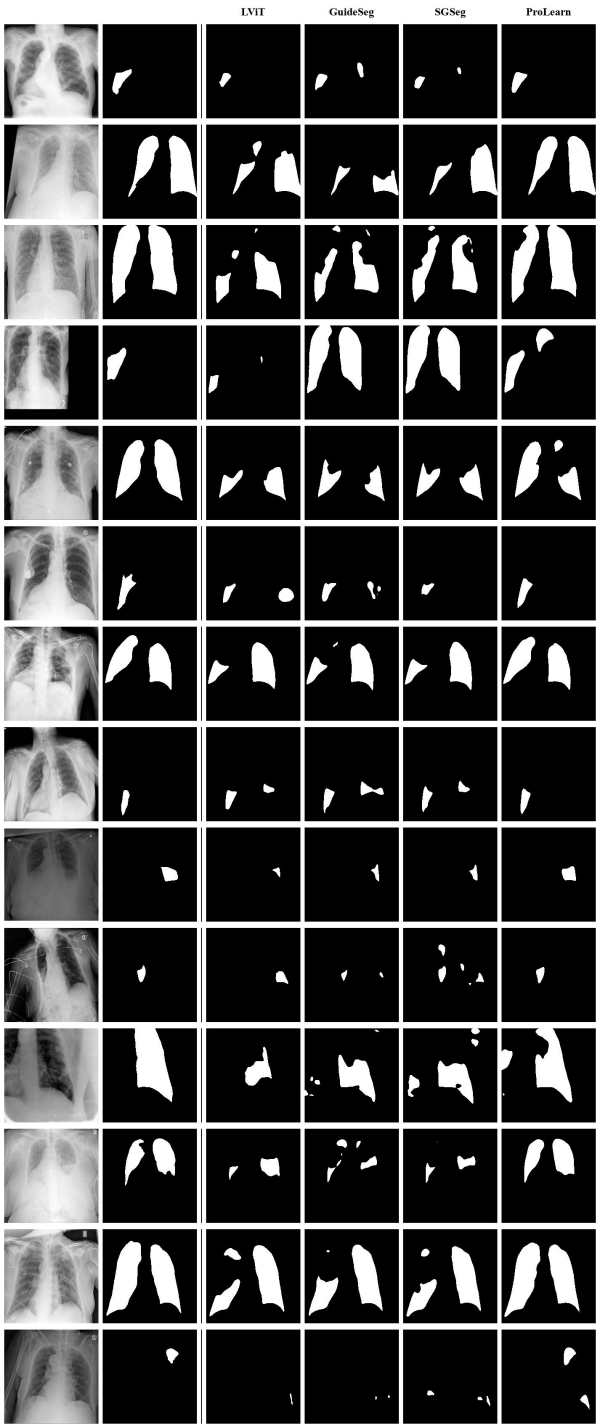

Figure A9. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on QaTa-COV19 dataset under 5% text availability.

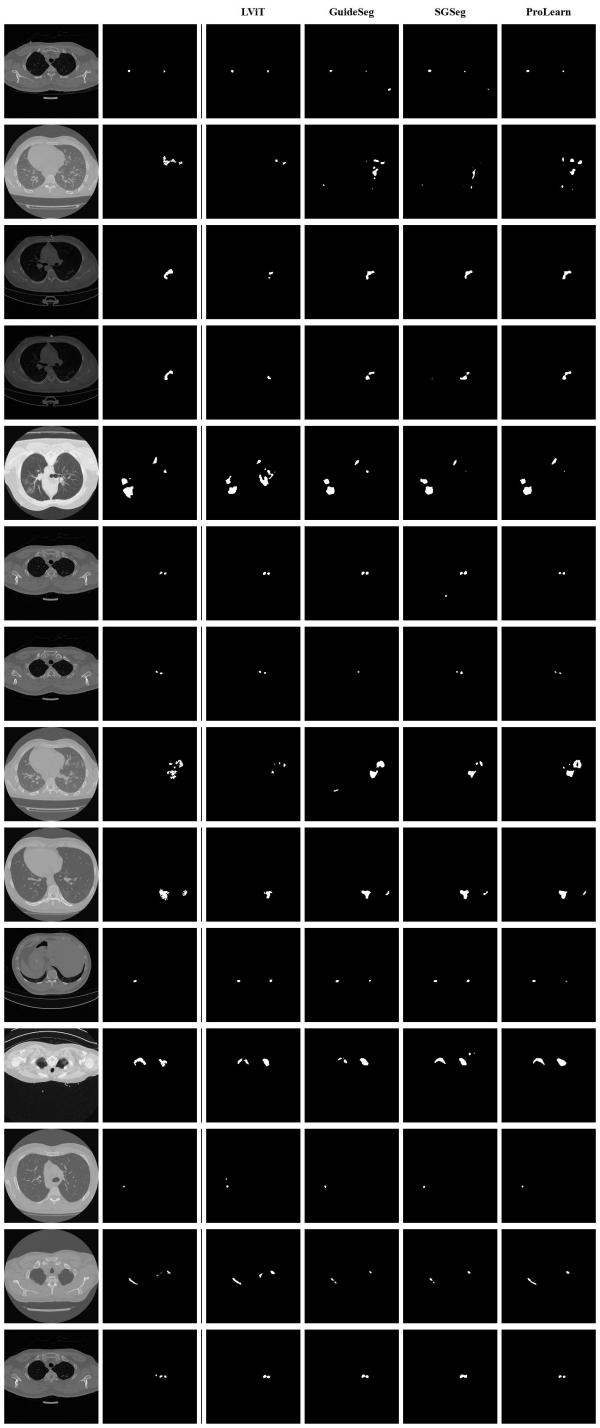

Figure A10. Comparison of segmentation results among LViT, GuideSeg, SGSeg, and our ProLearn on MosMedData+ dataset under 5% text availability.