# UniSeg: A Prompt-Driven Universal Segmentation Model as Well as A Strong Representation Learner

Yiwen Ye[1], Yutong Xie[2], Jianpeng Zhang[1], Ziyang Chen[1], and Yong Xia[1,3(✉)]

[1] National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, School of Computer Science and Engineering, Northwestern Polytechnical University, Xi'an 710072, China
yxia@nwpu.edu.cn

[2] Australian Institute for Machine Learning, The University of Adelaide, Adelaide, SA, Australia

[3] Ningbo Institute of Northwestern Polytechnical University, Ningbo 315048, China

**Abstract.** The universal model emerges as a promising trend for medical image segmentation, paving up the way to build medical imaging large model (MILM). One popular strategy to build universal models is to encode each task as a one-hot vector and generate dynamic convolutional layers at the end of the decoder to extract the interested target. Although successful, it ignores the correlations among tasks and meanwhile is too late to make the model 'aware' of the ongoing task. To address both issues, we propose a prompt-driven **Uni**versal **Seg**mentation model (UniSeg) for multi-task medical image segmentation using diverse modalities and domains. We first devise a learnable universal prompt to describe the correlations among all tasks and then convert this prompt and image features into a task-specific prompt, which is fed to the decoder as a part of its input. Thus, we make the model 'aware' of the ongoing task early and boost the task-specific training of the whole decoder. Our results indicate that the proposed UniSeg outperforms other universal models and single-task models on 11 upstream tasks. Moreover, UniSeg also beats other pre-trained models on two downstream datasets, providing the community with a high-quality pre-trained model for 3D medical image segmentation. Code and model are available at https://github.com/yeerwen/UniSeg.

**Keywords:** Prompt learning · Universal model · Medical image segmentation

Y. Ye and Y. Xie—Contributed equally.

**Supplementary Information** The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-43898-1_49.


H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14222, pp. 508–518, 2023.
https://doi.org/10.1007/978-3-031-43898-1_49

## 1 Introduction

Recent years have witnessed the remarkable success of deep learning in medical image segmentation. However, although the performance of deep learning models even surpasses the accuracy of human exports on some segmentation tasks, two challenges still persist. (1) Different segmentation tasks are usually tackled separately by specialized networks (see Fig. 1(a)), leading to distributed research efforts. (2) Most segmentation tasks face the limitation of a small labeled dataset, especially for 3D segmentation tasks, since pixel-wise 3D image annotation is labor-intensive, time-consuming, and susceptible to operator bias.

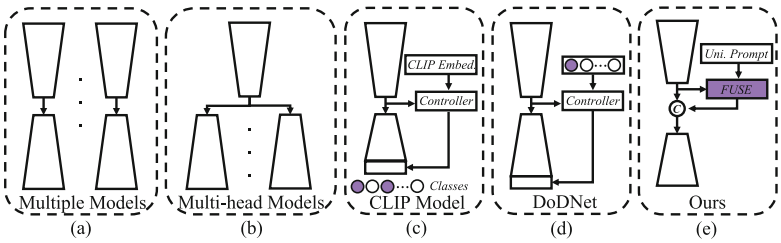


**Fig. 1.** Five strategies for multi-task medical image segmentation. (a) Multiple Models: Train $n$ models for $n$ tasks; (b) Multi-head Models: Train the model with one shared encoder and $n$ task-specific decoders; (c) CLIP-driven Model: Train one model on $n$ datasets by masking label-unavailable predictions; (d) Dynamic Convolution: Train one model on $n$ datasets using a one-hot vector as the task-related information; (e) Ours: Train one model on $n$ datasets using task-specific prompts. We use purple to highlight where to add the task-related information.

Several strategies have been attempted to address both challenges. First, multi-head networks (see Fig. 1(b)) were designed for multiple segmentation tasks [4,7,21]. A typical example is Med3D [4], which contains a shared encoder and multiple task-specific decoders. Although they benefit from the encoder parameter-sharing scheme and the rich information provided by multiple training datasets, multi-head networks are less-suitable for multi-task co-training, due to the structural redundancy caused by the requirement of preparing a separate decoder for each task. The second strategy is the multi-class model, which formulates multiple segmentation tasks into a multi-class problem and performs it simultaneously. To achieve this, the CLIP-driven universal model [16] (see Fig. 1(c)) introduces the text embedding of all labels as external knowledge, obtained by feeding medical prompts to CLIP [5]. However, CLIP has limited ability to generalize in medical scenarios due to the differences between natural and medical texts. It is concluded that the discriminative ability of text prompts is weak in different tasks, and it is difficult to help learn task-specific semantic information. The third strategy is dynamic convolution. DoDNet [29] and its variants [6,17,25] present universal models, which can perform different segmentation tasks based on using task encoding and a controller to generate dynamic

convolutions (see Fig. 1(d)). The limitations of these models are two-fold. (1) Different tasks are encoded as one-hot vectors, which are mutually orthogonal, ignoring the correlations among tasks. (2) The task-related information (*i.e.*, dynamic convolution parameters) is introduced at the end of the decoder. It may be too late for the model to be 'aware' of the ongoing task, making it difficult to decode complex targets.

In this paper, we propose a prompt-driven **Uni**versal **Seg**mentation model (UniSeg) to segment multiple organs, tumors, and vertebrae on 3D medical images with diverse modalities and domains. UniSeg contains a vision encoder, a fusion and selection (FUSE) module, and a prompt-driven decoder. The FUSE module is devised to generate the task-specific prompt, which enables the model to be 'aware' of the ongoing task (see Fig. 1(e)). Specifically, since prompt learning has a proven ability to represent both task-specific and task-invariant knowledge [24], a learnable universal prompt is designed to describe the correlations among tasks. Then, the universal prompt and the features extracted by the vision encoder are fed to the FUSE module to generate task prompts for all tasks. The task-specific prompt is selected according to the ongoing task. Moreover, to introduce the prompt information to the model early, we move the task-specific prompt from the end of the decoder to the start of the decoder (see Fig. 2). Thanks to both designs, we can use a single decoder and a segmentation head to predict various targets under the supervision of the corresponding ground truths. We collected 3237 volumetric data with three modalities (CT, MR, and PET) and various targets (eight organs, vertebrae, and tumors) from 11 datasets as the upstream dataset. On this dataset, we evaluated our UniSeg model against other universal models, such as DoDNet and the CLIP-driven universal model. We also compared UniSeg to seven advanced single-task models, such as CoTr [26], nnFormer [30], and nnUNet [12], which are trained independently on each dataset. Furthermore, to verify its generalization ability on downstream tasks, we applied the trained UniSeg to two downstream datasets and compared it to other pre-trained models, such as MG [31], DeSD [28], and UniMiSS [27]. Our results indicate that UniSeg outperforms all competing methods on 11 upstream tasks and two downstream tasks.

Our contributions are three-fold: (1) We design a universal prompt to describe the correlations among different tasks and use it to generate task prompts for all tasks. (2) We utilize the task-related prompt information as the input of the decoder, facilitating the training of the whole decoder, instead of just the last few layers. (3) The proposed UniSeg can be trained on and applied to various 3D medical image tasks with diverse modalities and domains, providing a high-quality pre-trained 3D medical image segmentation model for the community.

## 2 Method

### 2.1 Problem Definition

Let $\{D_1, D_2, ..., D_N\}$ be $N$ datasets. Here, $D_i = \{X_{ij}, Y_{ij}\}_{j=1}^{n_i}$ represents that the $i$-th dataset has a total of $n_i$ image-label pairs, and $X_{ij}$ and $Y_{ij}$ are the image

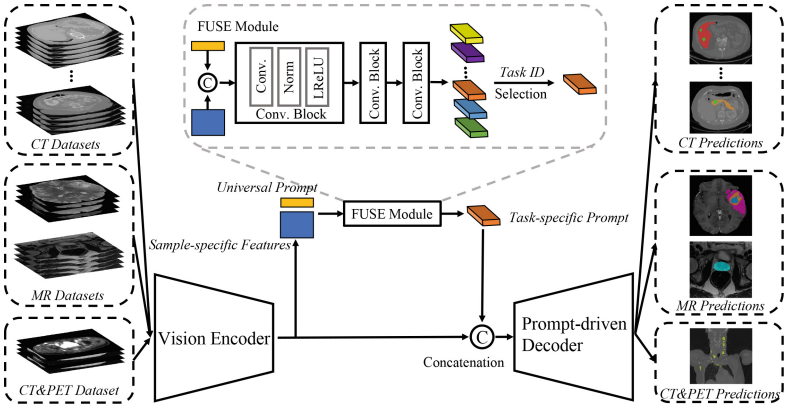


**Fig. 2.** Technical pipeline of our UniSeg, including a vision encoder, FUSE module, and a prompt-driven decoder. The sample-specific features produced by the encoder are concatenated with a learnable universal prompt as the input of the FUSE module. Then the FUSE module produces the task-specific prompt, which enables the model to be 'aware' of the ongoing task.

and the corresponding ground truth, respectively. Straightforwardly, $N$ tasks can be completed by training $N$ models on $N$ datasets, respectively. This solution faces the issues of (1) designing an architecture for each task, (2) distributing research effort, and (3) dropping the benefit of rich information from other tasks. Therefore, we propose a universal framework called UniSeg to solve multiple tasks with a single model, whose architecture was shown in Fig. 2.

### 2.2 Encoder-Decoder Backbone

The main architecture of UniSeg is based on nnUNet [12], which consists of an encoder and a decoder shared by different tasks. The encoder has six stages, each containing two convolutional blocks, to extract features and gradually reduce the feature resolution. The convolutional block includes a convolutional layer followed by instance normalization and a ReakyReLU activation, and the first convolution layer of each stage is usually set to reduce the resolution with a stride of 2, except for the first stage. To accept the multi-modality inputs, we reform the first convolution layer and set up three different convolution layers to handle the input with one, two, or four channels, respectively. After the encoder process, we obtain the sample-specific features $F \in \mathbb{R}^{C \times \frac{D}{16} \times \frac{H}{32} \times \frac{W}{32}}$, where $C$ is the number of channels and $D$, $H$, and $W$ are the depth, height, and width of the input, respectively. Symmetrically, in each stage of the decoder, the upsampling operation implemented by a transposed convolution layer is applied to the input feature map to improve its resolution and reduce its channel number. The upsampled feature map is concatenated with the output of the corresponding encoder stage and then fed to a convolutional block. After the decoder process,

the output of each decoder stage is passed through a segmentation head to predict segmentation maps for deep supervision, which is governed by the sum of the Dice loss and cross-entropy loss. Note that the channel number of multi-scale segmentation maps is set to the maximum number of classes among all tasks.

### 2.3 Universal Prompt

Following the simple idea that everything is correlated, we believe that the correlations among different segmentation tasks must exist undoubtedly, though they are ignored by DoDNet which uses a set of orthogonal and one-hot task codes. Considering the correlations among tasks are extremely hard to handcraft, we propose a learnable prompt called universal prompt to describe them and use that prompt to generate task prompts for all tasks, aiming to encourage interaction and fusion among different task prompts. We define the shape of the universal prompt as $F_{uni} \in \mathbb{R}^{N \times \frac{D}{16} \times \frac{H}{32} \times \frac{W}{32}}$, where $N$ is the number of tasks.

### 2.4 Dynamic Task Prompt

Before building a universal network, figuring out a way to make the model 'aware' of the ongoing task is a must. DoDNet adopts a one-hot vector to encode each task, and the CLIP-driven universal model [16] uses masked back-propagation to optionally optimize the task-related segmentation maps. By contrast, we first obtain $N$ features by passing the concatenation of $F_{uni}$ and $F$ through three convolutional blocks, shown as follows

$$\{F_{task1}, F_{task2}, ..., F_{taskN}\} = Split(f(cat(F_{uni}, F)))^N, \tag{1}$$

where $F_{taski}$ denotes the prompt features belonging to the $i$-th task, $cat(,)$ is a concatenation operation, $f(\cdot)$ denotes the feed forward process, and $Split(\cdot)^N$ means splitting features along the channel to obtain $N$ features with the same shape. Then, we select the target features, called task-specific prompt $F_{tp}$, from $\{F_{task1}, F_{task2}, ..., F_{taskN}\}$ according to the ongoing task. Finally, we concatenate $F$ and selected $F_{tp}$ as the decoder input. In this way, we introduce task-related prior information into the model, aiming to boost the training of the whole decoder rather than only the last few convolution layers.

### 2.5 Transfer Learning

After training UniSeg on upstream datasets, we transfer the pre-trained encoder-decoder and randomly initialized segmentation heads to downstream tasks. The model is fine-tuned in a fully supervised manner to minimize the sum of the Dice loss and cross-entropy loss.

# 3 Experiments and Results

## 3.1 Datasets and Evaluation Metric

**Datasets.** For this study, we collected 11 medical image segmentation datasets as the upstream dataset to train our UniSeg and single-task models. The Liver and Kidney datasets are from LiTS [3] and KiTS [11], respectively. The Hepatic Vessel (HepaV), Pancreas, Colon, Lung, and Spleen datasets are from Medical Segmentation Decathlon (MSD) [1]. VerSe20 [19], Prostate [18], BraTS21 [2], and AutoPET [8] datasets have annotations of the vertebrae, prostate, brain tumors, and whole-body tumors, respectively. We used the binary version of the VerSe20 dataset, where all foreground classes are regarded as one class. Moreover, we dropped the samples without tumors in the AutoPET dataset.

**Table 1.** Details of eleven upstream datasets and two downstream datasets.

| Dataset | Upstream | | | | | | | | | | | Downstream | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CT | | | | | | | | MR | | CT&PET | CT | MR |
| | Liver | Kidney | HepaV | Pancreas | Colon | Lung | Spleen | VerSe20 | Prostate | BraTS21 | AutoPET | BTCV | VS |
| Organ | ✓ | ✓ | ✓ | ✓ | × | × | ✓ | × | ✓ | × | × | ✓ | × |
| Tumor | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | × | × | × | ✓ | ✓ | × | ✓ |
| Vertebrae | × | × | × | × | × | × | × | ✓ | × | × | × | × | × |
| Train | 104 | 168 | 242 | 224 | 100 | 50 | 32 | 171 | 91 | 1000 | 400 | 21 | 193 |
| Test | 27 | 42 | 61 | 57 | 26 | 13 | 9 | 43 | 25 | 251 | 101 | 9 | 49 |

**Table 2.** Results of single-task models and universal models on eleven datasets. We use Dice (%) on each dataset and Mean Dice (%) on all datasets as metrics. The best results on each dataset are in bold.

| Method | Liver | Kidney | HepaV | Pancreas | Colon | Lung | Spleen | VerSe20 | Prostate | BraTS21 | AutoPET | Mean |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Single-task Model** | | | | | | | | | | | | |
| UNETR [9] | 62.6 | 69.9 | 53.8 | 44.1 | 6.0 | 56.0 | 94.2 | 86.0 | 85.3 | 83.5 | 62.2 | 64.0 |
| nnFormer [30] | 70.7 | 80.0 | 61.3 | 57.9 | 18.8 | 66.8 | 92.2 | 84.3 | 87.0 | 82.0 | 61.0 | 69.3 |
| PVTv2-B1 [23] | 67.7 | 83.8 | 65.1 | 59.6 | 39.8 | 68.5 | 95.3 | 84.7 | 88.5 | 83.4 | 61.4 | 72.5 |
| CoTr [26] | 74.7 | 85.1 | 67.2 | 65.8 | 33.8 | 66.9 | 95.2 | 87.1 | 88.0 | 82.9 | 58.8 | 73.2 |
| UXNet [15] | 75.4 | 82.2 | 67.3 | 59.4 | 39.8 | 59.5 | 95.7 | 87.1 | 88.8 | 84.3 | 68.2 | 73.4 |
| Swin UNETR [22] | 76.1 | 81.2 | 67.1 | 58.0 | 42.6 | 65.7 | 95.3 | 86.9 | 88.3 | 84.3 | 64.6 | 73.6 |
| nnUNet [12] | 77.2 | 87.5 | 69.6 | 68.8 | 49.0 | 68.4 | 96.2 | **87.2** | 89.4 | **84.4** | 64.6 | 76.6 |
| **Universal Model** | | | | | | | | | | | | |
| CLIP DoDNet | 62.1 | 83.6 | 57.0 | 53.3 | 19.6 | 43.8 | 51.4 | 80.2 | 89.3 | 83.1 | 65.6 | 62.6 |
| Universal Model [16] | 74.7 | 80.7 | 62.2 | 63.5 | 52.1 | 62.1 | 94.5 | 74.8 | 87.6 | 82.6 | 60.0 | 72.3 |
| DoDNet [29] | 76.7 | 87.2 | 70.4 | 70.5 | 54.6 | 69.9 | **96.5** | 86.1 | 89.1 | 83.2 | 65.3 | 77.2 |
| UniSeg | **79.1** | **88.2** | **71.2** | **70.9** | **55.0** | **70.9** | 96.4 | 86.1 | **89.7** | 83.3 | **69.4** | **78.2** |

Meanwhile, We use BTCV [14] and VS datasets [20] as downstream datasets to verify the ability of UniSeg to generalize to other medical image segmentation tasks. BTCV contains the annotations of 13 abdominal organs, including the spleen (Sp), right kidney (RKi), left kidney (LKi), gallbladder (Gb), esophagus (Es), liver (Li), stomach (St), aorta(Ao), inferior vena cava (IVC), portal vein

and splenic vein (PSV), pancreas (Pa), right adrenal gland (RAG), and left adrenal gland (LAG). The VS dataset contains the annotations of the vestibular schwannoma. More details are shown in Table 1.

**Evaluation Metric.** The Dice similarity coefficient (Dice) that measures the overlap region of the segmentation prediction and ground truth is employed to evaluate the segmentation performance.

**Table 3.** Results of self-supervised models and supervised pre-trained models. AutoPET and BraTS21 present the model per-trained on AutoPET and BraTS21 datasets, respectively. We use *italicized numbers* to indicate the performance gain using pre-trained weights. We repeat all experiments three times and report mean values.

| Dataset | BTCV | | | | | | | | | | | | | | VS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Sp | RKi | LKi | Gb | Es | Li | St | Ao | IVC | PSV | Pa | RAG | LAG | Mean | Tumor |
| MG [31] | 86.8 | 85.5 | 83.0 | 63.5 | 70.5 | 92.4 | 78.3 | 88.5 | 85.3 | 70.7 | 71.4 | 68.7 | 58.2 | 77.1 *+2.7* | 79.3 *+7.2* |
| GVSL [10] | 90.6 | 92.3 | 91.2 | 63.7 | 72.5 | 95.6 | 80.1 | 87.5 | 84.4 | 71.7 | 72.7 | 68.1 | 63.6 | 79.5 *+1.9* | 91.0 *+2.2* |
| SMIT [13] | 90.7 | 92.1 | 91.9 | 63.0 | 74.8 | 95.7 | 75.9 | 88.6 | 86.4 | 72.8 | 74.3 | 71.3 | 69.5 | 80.6 *+1.3* | 92.2 *+2.3* |
| UniMiSS [27] | 95.0 | 92.9 | 91.5 | 67.1 | 73.6 | 96.4 | 82.4 | 88.9 | 83.9 | 73.2 | 76.2 | 67.1 | 67.0 | 81.2 *+3.0* | 91.4 *+2.0* |
| DeSD [28] | 96.1 | 94.6 | 93.2 | 64.4 | 75.2 | 96.6 | 88.7 | 90.0 | 87.5 | 75.1 | 79.9 | 70.4 | 70.5 | 83.3 *+0.8* | 92.2 *+1.5* |
| AutoPET | 95.5 | 93.4 | 91.4 | 62.8 | 75.3 | 96.5 | 84.6 | 90.0 | 87.2 | 75.5 | 79.4 | 71.2 | 71.3 | 82.6 *−0.5* | 91.1 *+0.4* |
| BraTS21 | 95.9 | 93.4 | 90.8 | 69.2 | 76.5 | 96.6 | 84.9 | 90.2 | 87.6 | 76.0 | 80.8 | 72.4 | 71.7 | 83.5 *+0.4* | 91.2 *+0.4* |
| DoDNet [29] | 96.4 | 94.5 | 89.7 | 68.3 | 76.9 | 96.8 | 86.5 | 89.8 | 87.7 | 76.1 | 81.9 | 73.2 | 75.2 | 84.1 *+0.9* | 91.8 *+1.1* |
| UniSeg | 96.2 | 94.4 | 91.6 | 68.4 | 77.9 | 96.7 | 87.8 | 90.1 | 87.6 | 76.7 | 83.3 | 73.4 | 75.1 | 84.6 *+1.4* | 92.9 *+2.1* |

### 3.2 Implementation Details

Both pre-training on eleven upstream datasets and fine-tuning on two downstream datasets were implemented based on the nnUNet framework [12]. During pre-training, we adopted the SGD optimizer and set the batch size to 2, the initial learning rate to 0.01, the default patch size to $64 \times 192 \times 192$, and the maximum training epoch to 1000 with a total of 550,000 iterations. Moreover, we adopted a uniform sampling strategy to sample training data from upstream datasets. In the inference stage, we employed the sliding window strategy, in which the shape of the window is the same as the training patch size, to obtain the whole average segmentation map. During fine-tuning, We set the batch size to 2, the initial learning rate to 0.01, the default patch size to $48 \times 192 \times 192$, and the maximum training iterations to 25,000 for all downstream datasets. The sliding window strategy was also employed when inference on downstream tasks.

### 3.3 Results

**Comparing to Single-Task and Universal Models.** Our UniSeg was compared with advanced single-task models and universal models. The former includes UNETR [9], nnFormer [30], PVTv2-B1 [23], CoTr [26], UXNet [15], Swin UNETR [22], and nnUNet [12]. The latter includes DoDNet [29], CLIP

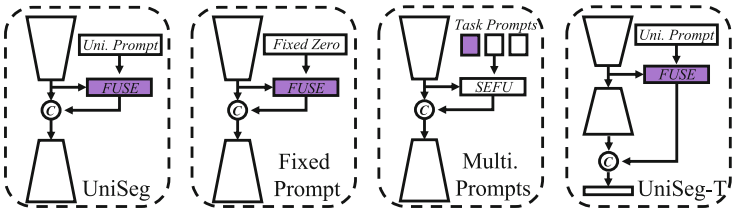


**Fig. 3.** Diagram of UniSeg, Fixed Prompt, Multiple Prompts, and UniSeg-T. Fixed Prompt initializes a zero prompt with no update. Multiple Prompts adopts multiple task-specific prompts. UniSeg-T adds the task-related prompt at the end of the decoder. We use purple to highlight where to add the task-related information.

**Table 4.** Results of baseline, Fixed Prompt, Multiple Prompts, UniSeg-T, and our UniSeg. The baseline means the performance of our encoder-decoder backbone respectively trained on each dataset. We compare the mean Dice (%) of eleven datasets.

| Method | Baseline | Fixed Prompt | Multi. Prompts | UniSeg-T | UniSeg |
|---|---|---|---|---|---|
| Dice | 76.6 | 77.4 | 77.5 | 76.9 | 78.2 |

DoDNet, which replaces the one-hot vectors with CLIP embeddings obtained by following [16], and CLIP-driven universal model [16]. For a fair comparison, the maximum training iterations of single-task models on each task are 50,000, and the patch size is $64 \times 192 \times 192$, except for Swin UNETR, whose patch size is $64 \times 160 \times 160$ due to the limitation of GPU memory. The backbones of the competing universal models and our UniSeg are the same. As shown in Table 2, Our UniSeg achieves the highest Dice on eight datasets, beating the second-best models by 1.9%, 0.7%, 0.8%, 0.4%, 0.4%, 1.0%, 0.3%, 1.2% on the Liver, Kidney, HepaV, Pancreas, Colon, Lung, Prostate, and AutoPET datasets, respectively. Moreover, UniSeg also presents superior performance with an average margin of 1.0% and 1.6% on eleven datasets compared to the second-best universal model and single-task model, respectively, demonstrating its superior performance.

**Comparing to Other Pre-trained Models.** We compared our UniSeg with advanced unsupervised pre-trained models, such as MG [31], SMIT [13], UniMiSS [27], DeSD [28], and GVSL[10], and supervised pre-trained models, such as AutoPET and DoDNet [29]. The former are officially released with different backbones while the latter are trained using the datasets and backbone used in our UniSeg. To verify the benefit of training on multiple datasets, we also report the performance of the models per-trained on AutoPET and BraTS21, respectively. The results in Table 3 reveal that almost all pre-trained models achieve performance gains over their baselines, which were trained from scratch. More important, thanks to the powerful baseline and small gap between the pretext and downstream tasks, UniSeg achieves the best performance and competitive performance gains on downstream datasets, demonstrating that it has learned a strong representations ability. Furthermore, another advantage of UniSeg against

other unsupervised pre-trained models is that it is more resource-friendly, requiring only one GPU of 11 GB memory for implementation, while unsupervised pre-trained models usually require tremendous computational resources, such as eight and four V100 for UniMiSS and SMIT, respectively.

**Comparison of Different Variants.** We attempted three UniSeg variants, including Fixed Prompt, Multiple Prompts, and UniSeg-T, as shown in Fig. 3. The results in Table 4 suggest that (1) learnable universal prompt is helpful for building valuable prompt features; (2) using one universal prompt instead of multiple task-independent prompts boosts the interaction and fusion among all tasks, resulting in better performance; (3) adding task-related information in advance facilitates handling complex prediction situations.

## 4 Conclusion

This study proposes a universal model called UniSeg (a single model) to perform multiple organs, tumors, and vertebrae segmentation on images with multiple modalities and domains. To solve two limitations existing in preview universal models, we design the universal prompt to describe correlations among all tasks and make the model 'aware' of the ongoing task early, boosting the training of the whole decoder instead of just the last few layers. Thanks to both designs, our UniSeg achieves superior performance on 11 upstream datasets and two downstream datasets, setting a new record. In our future work, we plan to design a universal model that can effectively process multiple dimensional data.

**Acknowledgements.** This work was supported in part by the Key Research and Development Program of Shaanxi Province, China, under Grant 2022GY-084, in part by the National Natural Science Foundation of China under Grant 62171377, in part by the Ningbo Clinical Research Center for Medical Imaging under Grant 2021L003 (Open Project 2022LYKFZD06), and in part by the Innovation Foundation for Doctor Dissertation of Northwestern Polytechnical University under Grant CX2022056.

## References

1. Antonelli, M., et al.: The medical segmentation decathlon. arXiv preprint arXiv:2106.05735 (2021)
2. Baid, U., et al.: The RSNA-ASNR-MICCAI BraTS 2021 benchmark on brain tumor segmentation and radiogenomic classification. arXiv preprint arXiv:2107.02314 (2021)
3. Bilic, P., et al.: The liver tumor segmentation benchmark (LiTs). arXiv preprint arXiv:1901.04056 (2019)
4. Chen, S., Ma, K., Zheng, Y.: Med3D: transfer learning for 3D medical image analysis. arXiv preprint arXiv:1904.00625 (2019)
5. Conneau, A., Lample, G.: Cross-lingual language model pretraining. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
6. Deng, R., Liu, Q., Cui, C., Asad, Z., Yang, H., Huo, Y.: Omni-Seg: a single dynamic network for multi-label renal pathology image segmentation using partially labeled data. arXiv preprint arXiv:2112.12665 (2021)

7. Fang, X., Yan, P.: Multi-organ segmentation over partially labeled datasets with multi-scale feature abstraction. IEEE Trans. Med. Imaging **39**(11), 3619–3629 (2020)
8. Gatidis, S., et al.: A whole-body FDG-PET/CT dataset with manually annotated tumor lesions. Sci. Data **9**(1), 601 (2022)
9. Hatamizadeh, A., et al.: UNETR: transformers for 3D medical image segmentation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 574–584 (2022)
10. He, Y., et al.: Geometric visual similarity learning in 3D medical image self-supervised pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
11. Heller, N., et al.: The state of the art in kidney and kidney tumor segmentation in contrast-enhanced CT imaging: results of the KiTS19 challenge. Med. Image Anal. **67**, 101821 (2021)
12. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nat. Methods **18**(2), 203–211 (2021)
13. Jiang, J., Tyagi, N., Tringale, K., Crane, C., Veeraraghavan, H.: Self-supervised 3D anatomy segmentation using self-distilled masked image transformer (SMIT). In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) Medical Image Computing and Computer Assisted Intervention – MICCAI 2022. MICCAI 2022. LNCS, vol. 13434, pp. 556–566. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16440-8_53
14. Landman, B., Xu, Z., Igelsias, J., Styner, M., Langerak, T., Klein, A.: MICCAI multi-atlas labeling beyond the cranial vault-workshop and challenge. In: Proceedings of MICCAI Multi-Atlas Labeling Beyond Cranial Vault-Workshop Challenge, vol. 5, p. 12 (2015)
15. Lee, H.H., Bao, S., Huo, Y., Landman, B.A.: 3D UX-Net: a large Kernel volumetric convnet modernizing hierarchical transformer for medical image segmentation. In: The Eleventh International Conference on Learning Representations (2023)
16. Liu, J., et al.: Clip-driven universal model for organ segmentation and tumor detection. arXiv preprint arXiv:2301.00785 (2023)
17. Liu, P., et al.: Universal segmentation of 33 anatomies. arXiv preprint arXiv:2203.02098 (2022)
18. Liu, Q., Dou, Q., Yu, L., Heng, P.A.: MS-Net: multi-site network for improving prostate segmentation with heterogeneous MRI data. IEEE Trans. Med. Imaging **39**(9), 2713–2724 (2020)
19. Sekuboyina, A., et al.: VERSE: a vertebrae labelling and segmentation benchmark for multi-detector ct images. Med. Image Anal. **73**, 102166 (2021)
20. Shapey, J., et al.: Segmentation of vestibular schwannoma from magnetic resonance imaging: an open annotated dataset and baseline algorithm. The Cancer Imaging Archive (2021)
21. Shi, G., Xiao, L., Chen, Y., Zhou, S.K.: Marginal loss and exclusion loss for partially supervised multi-organ segmentation. Med. Image Anal. **70**, 101979 (2021)
22. Tang, Y., et al.: Self-supervised pre-training of Swin transformers for 3D medical image analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20730–20740 (2022)
23. Wang, W., et al.: PVT v2: improved baselines with pyramid vision transformer. Comput. Vis. Media **8**(3), 415–424 (2022)

24. Wang, Z., et al.: Learning to prompt for continual learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 139–149 (2022)
25. Wu, H., Pang, S., Sowmya, A.: Tgnet: a task-guided network architecture for multi-organ and tumour segmentation from partially labelled datasets. In: International Symposium on Biomedical Imaging, pp. 1–5. IEEE (2022)
26. Xie, Y., Zhang, J., Shen, C., Xia, Y.: CoTr: efficiently bridging CNN and transformer for 3D medical image segmentation. In: de Bruijne, M., et al. (eds.) MICCAI 2021. LNCS, vol. 12903, pp. 171–180. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-87199-4_16
27. Xie, Y., Zhang, J., Xia, Y., Wu, Q.: UniMiss: universal medical self-supervised learning via breaking dimensionality barrier. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. ECCV 2022. LNCS, vol. 13681, pp. 558–575. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-19803-8_33
28. Ye, Y., Zhang, J., Chen, Z., Xia, Y.: DeSD: self-supervised learning with deep self-distillation for 3D medical image segmentation. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) Medical Image Computing and Computer Assisted Intervention – MICCAI 2022. MICCAI 2022. LNCS, vol. 13434, pp. 545–555. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16440-8_52
29. Zhang, J., Xie, Y., Xia, Y., Shen, C.: DoDNet: learning to segment multi-organ and tumors from multiple partially labeled datasets. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1195–1204 (2021)
30. Zhou, H.Y., Guo, J., Zhang, Y., Yu, L., Wang, L., Yu, Y.: nnFormer: interleaved transformer for volumetric segmentation. arXiv preprint arXiv:2109.03201 (2021)
31. Zhou, Z., Sodha, V., Pang, J., Gotway, M.B., Liang, J.: Models genesis. Med. Image Anal. **67**, 101840 (2021)