

Conditioned and composed image retrieval combining and partially fine-tuning CLIP-based features

Alberto Baldrati, Marco Bertini, Tiberio Uricchio, Alberto Del Bimbo

摘要

本文提出了一种基于 CLIP 预训练模型的条件和组合图像检索方法, 这是在基于内容的图像检索 (CBIR) 上的扩展, 图像与用户提供的修改意图文本信息相结合, 能在电子商务等领域得到广泛应用。该方法分为两个阶段, 第一阶段使用了视觉和文本特征的简单组合来微调 CLIP 文本编码器, 接着第二阶段使用了一种更复杂的 Combiner network 的结构来对视觉和文本特征进行融合, 两个阶段都用了对比学习的方法。所提出的方法在 FashionIQ 数据集和 CIRR 数据集上获得了条件 CBIR 的最好性能^[1]。

关键词: CLIP; 图像检索; 对比学习

1 引言

图像检索是计算机视觉的一个研究热点, 传统的图像检索系统通常以文本或者图像作为输入, 也就是基于文本的图像检索 (TBIR) 和基于内容的图像检索 (CBIR)。但是纯文本或单一图像通常无法准确表达用户的意图, 在实际应用场景中, 很多时候用户希望根据已有的图像增加自己的修改意见来查询目标图像, 即用一张查询图像和一个包含修改信息的文本来检索符合条件的目标图像, 图 1 就是一个具体的例子。通过添加文本信息来帮助 CBIR 系统进行图像检索, 这样做的目的是为了克服所使用的低级视觉特征和图像的高级含义之间的语义鸿沟。



图 1: 基于多模态特征融合的图像检索

2 相关工作

在过去的几年里, 一些调查概述了 CBIR 方法及其演变。Zheng 等人^[2]和 Zhou 等人^[3]调查了包括基于工程和基于学习特征的图像搜索方法。最近, Dubey^[4]调查了基于深度学习的 CBIR 方法。

2.1 视觉和语言预训练

OpenAI CLIP 网络模型^[5]最近在多模态零样本学习的任务中获得显著的结果, 简单来说, 尽管没有直接针对特定基准进行优化, 但由于图像和文本的泛化能力, 它在不同的任务上表现良好。CLIP 使用从网络上抓取的 400 亿个文本图像对来学习图像和文本描述之间的关联进行训练。

2.2 条件时尚图像检索

这项工作涉及最近的条件时尚图像检索问题^[6],这个任务已经在大量的著作中得到了解决。在^[7]中,提出了一种基于转换器的方法,该方法可以无缝地插入到 CNN 中,以选择性地保留和转换以语言语义为条件的视觉特征。在^[8]中,已经提出了文本图像残差门控 (TIRG),这是一种使用门控和残差特征结合图像和文本特征的方法。在^[9]中,已经提出了 ComposeAE,这是一种基于自动编码器的模型,用于使用深度学习 (DML) 方法学习图像和文本特征的组合。在^[10]中,已经提出使用一种称为 CurlingNet 的方法来衡量图像相对于条件查询文本之间的语义关系。条件图像检索最近扩展到^[11]中的多轮对话。该系统使用 ComposeAE^[9] 在每一轮组合图像和文本,根据轮次顺序将其输入循环网络。

3 本文方法

3.1 本文方法概述

本文提出了一个基于 CLIP 预训练模型的图像检索二阶段训练方法,首先在第一阶段微调 CLIP 预训练模型的文本编码器模块,该阶段的文本和图像特征通过简单的加权求和进行组合,然后在第二阶段训练一个 Combiner network 进行文本和图像特征的组合。

3.2 第一阶段：微调 CLIP 编码器模块

图 2 是第一阶段训练过程的概述图,首先将文本信息和参考图像数据输入 CLIP 预训练模型,经过 CLIP 处理得到文本和图像特征,然后将此进行简单的加权求和,得到一个组合特征,然后将其与目标图像经过 CLIP 图像编码器处理得到的图像特征进行对比学习,更新 CLIP 文本编码器的权重。

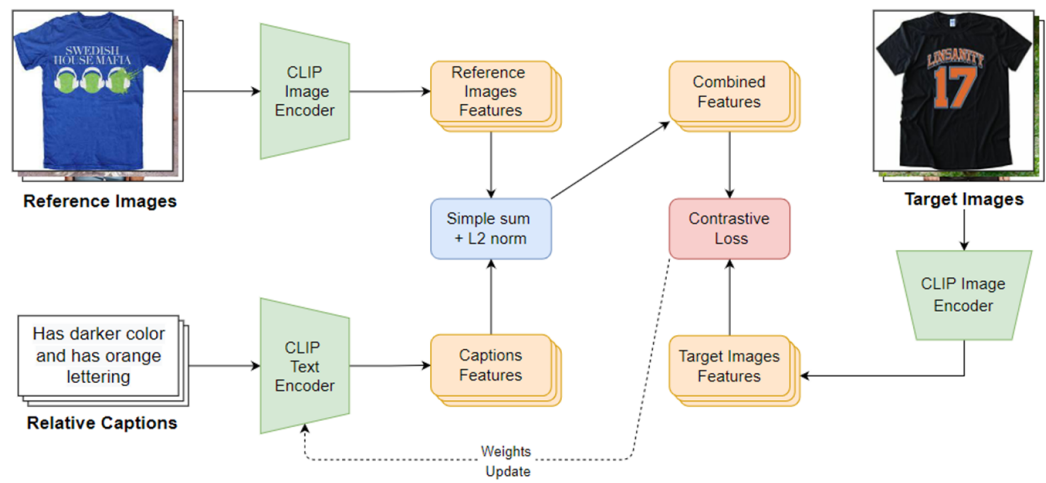


图 2: 第一阶段训练过程概述

3.3 第二阶段：Combiner network 模块

图 3 是第二阶段训练过程的概述图,该阶段将冻结 CLIP 文本和图像编码器部分,然后将第一阶段的简单加权求和改为 Combiner network 对经过 CLIP 处理的文本和图像特征进行组合,再使用跟第一阶段相同的对比学习的方法更新 Combiner network 的权重进行训练。图 4 是 Combiner network 的具体结构图。

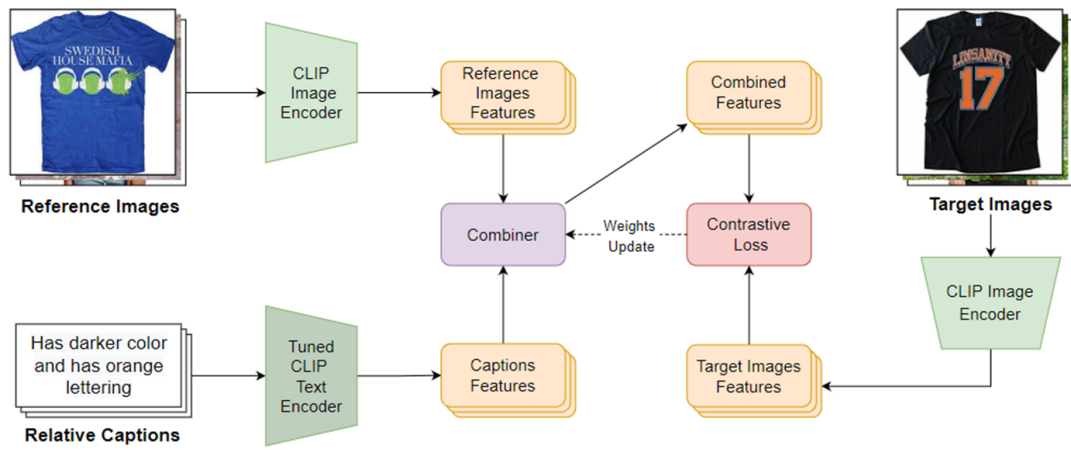


图 3: 第二阶段训练过程概述

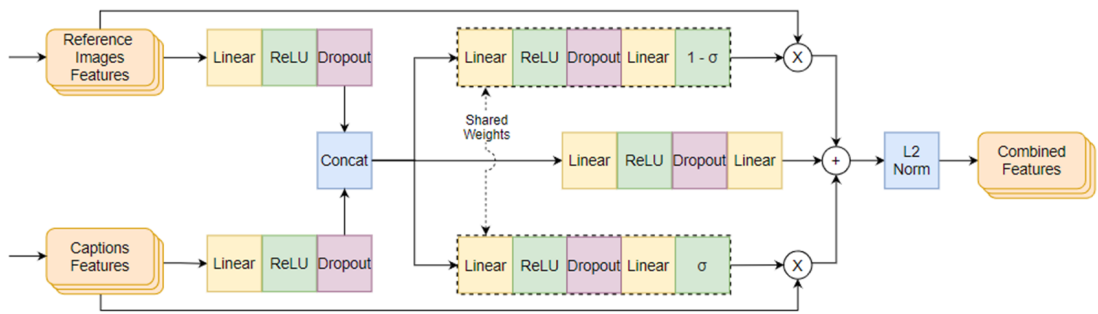


图 4: Combiner network 的结构图

4 复现细节

4.1 与已有开源代码对比

本次复现工作的代码不完全，参考了论文作者给出的代码，在这基础上对代码中的数据集加载模块、优化训练进度以及微调预训练的方法进行了添加和修改，完成了对论文中两阶段训练方法的复现工作。图 5 是第一阶段和第二阶段的训练方法代码。

```
def clip_finetune_fiq(train_dress_types: List[str], val_dress_types: List[str],
                     num_epochs: int, clip_model_name: str, learning_rate: float, batch_size: int,
                     validation_frequency: int, transform: str, save_training: bool, encoder: str, save_best: bool,
                     **kwargs):

def combiner_training_fiq(train_dress_types: List[str], val_dress_types: List[str],
                          projection_dim: int, hidden_dim: int, num_epochs: int, clip_model_name: str,
                          combiner_lr: float, batch_size: int, clip_bs: int, validation_frequency: int,
                          transform: str, save_training: bool, save_best: bool, **kwargs):
```

图 5: 实现第一、二阶段训练的方法代码

4.2 加载数据集代码模块

由于论文作者提供的代码不包括数据集加载部分，所以这一部分的代码需要我自己进行补充，才能将收集到的数据集载入到网络中进行训练。图 6 是加载数据集模块和数据集划分的部分代码。

```

if self.split == 'train':
    reference_image_path = data_path + '/fashionIQ_dataset/images/' + f"{reference_name}.jpg"
    reference_image = self.preprocess(PIL.Image.open(reference_image_path))
    target_name = self.triplets[index]['target']
    target_image_path = data_path + '/fashionIQ_dataset/images/' + f"{target_name}.jpg"
    target_image = self.preprocess(PIL.Image.open(target_image_path))
    return reference_image, target_image, image_captions

elif self.split == 'val':
    target_name = self.triplets[index]['target']
    return reference_name, target_name, image_captions

elif self.split == 'test':
    reference_image_path = data_path + '/fashionIQ_dataset/images/' + f"{reference_name}.jpg"
    reference_image = self.preprocess(PIL.Image.open(reference_image_path))
    return reference_name, reference_image, image_captions

```

图 6: 加载数据集模块的代码

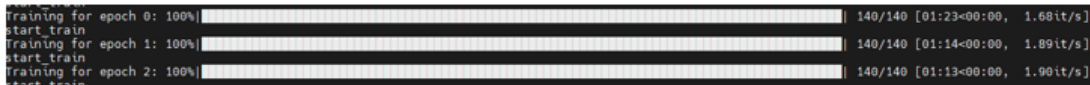
4.3 优化训练进度

这是对训练过程的显示进度条进行了修改，由于原本的代码调用了需要一个需要服务器联网才能显示训练进度的包，而为了避免联网带来的不必要麻烦，对训练过程的进度条进行了一点修改，使训练进度可视化。图 7 是修改的部分代码和修改后的训练进度显示。

```

train_running_results = {'images_in_epoch': 0, 'accumulated_train_loss': 0}
train_bar = tqdm(relative_train_loader, desc='Training for epoch ' + str(epoch))
#train_bar = tqdm(relative_train_loader, ncols=150)

```



```

Training for epoch 0: 100% | 140/140 [01:23<00:00, 1.68it/s]
start train
Training for epoch 1: 100% | 140/140 [01:14<00:00, 1.89it/s]
start train
Training for epoch 2: 100% | 140/140 [01:13<00:00, 1.90it/s]
start train

```

图 7: 修改后的训练进度显示

4.4 创新点

通过一系列的消融实验得到了一个与论文第一阶段相近的结果，但还是差了一点，然后在这基础上进行了网络训练方法的改进，在微调训练 CLIP 文本和图像编码器时，冻结 CLIP 图像编码器 BN 层的权重。经过方法的调整后，在第一阶段的训练结果也终于比论文的结果要好。

5 实验结果分析

这一章节将对实验结果进行分析，首先是第一阶段的复现结果：论文提出的方法说的是仅对 CLIP 文本编码器进行微调的效果是最好的，但我复现的仅对文本编码器进行微调的结果跟论文的结果差距很大，因此我进行了消融实验，发现同时对文本和图像编码器进行微调的效果比仅对文本编码器进行微调的效果要好，而且在 FashionIQ 数据集的 Shirt 类上的结果比论文的还要好上许多，但平均结果还是差了一点，然后我在此基础上在微调时加上对图像编码器的 BN 层进行冻结操作，效果得到进一步提高，也略微超过了论文的结果。图 5 为复现第一阶段的一些实验结果（第一行为论文上的结果）。

	FT	CF	<u>Batchsize</u>	Shirt	Dress	<u>Toptee</u>	Average
	Text-only	Sum	128	52.21	51.66	58.95	54.27
ours	Text-only	Sum	128	49.56	39.81	50.59	46.65
	Text-only	Sum	64	50.10	38.42	50.58	46.37
	Text-only	Sum	256	49.66	40.01	50.89	46.85
	Both	Sum	128	58.10	43.88	57.31	53.10
	Both(BN)	Sum	128	59.32	45.71	58.64	54.29

图 8: 第一阶段的实验结果

第二阶段就是在第一阶段微调训练结果的基础上将 Sum 改为论文提出的 Combiner network 结构进行训练，得到的结果有所提高但并非如论文所说的提升那么大，所以最后的结果和论文的结果相比还是有差距的。图 6 为复现第二阶段的一些实验结果（第一第二行为论文上的结果）。

	FT	CF	<u>Batchsize</u>	Shirt	Dress	<u>Toptee</u>	Average
	None	Combiner	4096	53.38	51.31	57.01	53.90
	Text-only	Combiner	4096	57.02	56.02	62.77	58.60
ours	None	Combiner	4096	51.76	38.18	47.88	45.94
	Text-only	Combiner	4096	55.05	41.20	52.57	49.60
	Both	Combiner	4096	60.06	45.41	59.87	55.11
	Both(BN)	Combiner	4096	61.15	47.09	61.32	56.52

图 9: 第二阶段的实验结果

6 总结与展望

通过这次的复现工作，算是真正入门了研究领域，了解到了复现一篇论文需要的工作，也学会了在现有工作的基础上去发现问题和解决问题，后续的话还会继续关注这篇文章的相关后续工作，然后看能不能将 Combiner network 进行改进提点。

参考文献

- [1] BALDRATI A, BERTINI M, URICCHIO T, et al. Conditioned and composed image retrieval combining and partially fine-tuning CLIP-based features[J]. 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2022: 4955-4964.
- [2] ZHENG L, YANG Y, TIAN Q. SIFT Meets CNN: A Decade Survey of Instance Retrieval[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018.
- [3] ZHOU W, LI H, TIAN Q. Recent Advance in Content-based Image Retrieval: A Literature Survey[J]. arXiv: Multimedia, 2017.
- [4] DUBEY S R. A Decade Survey of Content Based Image Retrieval using Deep Learning[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2021.

- [5] RADFORD A, KIM J W, HALLACY C, et al. Learning Transferable Visual Models From Natural Language Supervision[J]. international conference on machine learning, 2021.
- [6] WU H, GAO Y, GUO X, et al. Fashion IQ: A New Dataset Towards Retrieving Images by Natural Language Feedback[J]. computer vision and pattern recognition, 2019.
- [7] CHEN Y, GONG S, BAZZANI L. Image Search With Text Feedback by Visiolinguistic Attention Learning[J]. computer vision and pattern recognition, 2020.
- [8] VO N, JIANG L, SUN C, et al. Composing Text and Image for Image Retrieval - An Empirical Odyssey [J]. computer vision and pattern recognition, 2018.
- [9] ANWAAR M U, LABINTCEV E, KLEINSTEUBER M. Compositional Learning of Image-Text Query for Image Retrieval[J]. workshop on applications of computer vision, 2020.
- [10] YU Y, LEE S H, CHOI Y, et al. CurlingNet: Compositional Learning between Images and Text for Fashion IQ Data.[J]. arXiv: Computer Vision and Pattern Recognition, 2020.
- [11] YUAN Y, LAM W. Conversational Fashion Image Retrieval via Multiturn Natural Language Feedback [J]. international acm sigir conference on research and development in information retrieval, 2021.