Report to Cross-Modal Image Retrieval Track of 1st Foundation Model Challenge

Zhenghai He* Fuzhi Duan* Jun Lin Yanxun Yu Yayun Wang Zhongbin Niu
Xingmeng Hao Youxian Zheng Zhijiang Du
ZheJiang Dahua Technology CO., LTD, Hangzhou, China
{he_zhenghai,duan_fuzhi}@dahuatech.com

Abstract

Cross-modal image retrieval has become mainstream with the development of large multimodal model technology. Cross-Modal Image Retrieval Track of 1st Foundation Model Challenge is launched to research cross-modal image retrieval performance on traffic scenes. In this report, we describe the technical details of our submission to the challenge. CLIP [6] is introduced as an overall network architecture at first. Then we make adversarial attack to improve the robustness of the network. In the training stage, the Exponential Moving Average (EMA) significantly enhances training stability. Test Time Augmentation (TTA) and model fusion are applied in the test stage to improve the evaluation metrics. With these methods, we achieve Rank 2 in the challenge leaderboard B. The code is available at https://aistudio.baidu.com/aistudio/projectdetail/6210965.

1. Introduction

Video surveillance data is growing rapidly with the popularization of surveillance devices, which has led to increasing demand for effective analysis of the data. The high-performance image retrieval capability in traffic scenes is more and more crucial for traffic law enforcement and public security governance. Traditional image retrieval methods pay more attention to searching for images by image. Nowadays, cross-modal retrieval has become mainstream.

Cross-Modal Image Retrieval Track constructs a text retrieval image dataset with two categories of traffic participants: pedestrians and vehicles. The dataset has a total of 153728 images, including 136117 images in the training set and 17611 images in the validation set. The data distribution is shown in Table 1. There is one label for every image in the dataset, which is consisted of image name, attribute annotation, and text. Attribute annotation contains categories of participants, and text is a sentence describing the attribute. In addition, the attribute of pedestrians is much richer than that of vehicles. Some examples are shown in Fig. 1. Moreover, Complicated scenes and generalization (leaderboard A/B) make it more challenging for the competition. To address these challenges, we make several solid improvements based on CLIP. The results of Leaderboard B demonstrate the robustness and transferability of our solution. The implementation details mentioned above are illustrated in section 2 and section 3.

(a) (b) (c)

Figure 1. Images of the training dataset. There are two types of annotations for vehicles. For Instance, (a) is labeled as 'White Audi$This is a white Audi.', while (b) is labeled as 'Minivan$This is a Minivan.'. Besides, the annotation of pedestrians is much more complex than vehicles. (c) is labeled as '1,0,1,0,1,0,0,0,0,0,1,0,1,1,0,0,0,0,0,0,0$A pedestrian who is female is an adult person aged 18-60, with her body facing the camera and has a shoulderbag. She is in a shirt with short sleeves.'.

Table 1. Dataset distribution. The number of pedestrians is almost twice as that of vehicles.

Category	Training Set	Test Set
pedestrians	90000	10000
vehicles	46117	7611
sum	136117	17611

2. Method

Our solution is built on the foundation of an elaborate analysis of the dataset. We notice that annotations of both pedestrians and vehicles possess notable features. As for

*Equal contribution

vehicles, all attributes can be summarized into three categories: color, brand, and type. Furthermore, vehicle attribute annotations in the training and validation dataset are incomplete, while every query text in the test dataset specifies all three vehicle attributes, as shown in Table 2. When it comes to pedestrians, the attribute annotation is much more various. Apart from that, all attributes are hidden through binary values deliberately.

Table 2. Examples of vehicle retrieval text in the training phase and test phase. Only in the test phase, all three vehicle attributes of color, brand, and type are included.

Phase	Text
training case 1	This is a white JMC.
training case 2	This is a Minivan.
test case	A white JMC Minivan.

Due to the inconsistency in the dataset, it is hard to solve the task by classification network. As a result, we develop our solution based on CLIP. The overall network architecture is presented in Fig. 2. Following CLIP, we jointly train an image encoder [7] (vision transformer) and a text encoder (text transformer) to predict the correct pairings of a batch of (image, text) training examples. Given that data of Leaderboard B is not released, we make use of several methods to improve the performance and generalization of the model.

Data Augmentation Strong augmentation strategies are applied to images to enhance the performance. The augmentation pipeline includes random resize crop, random horizontal flip, random color jitter, random affine, and AutoAugment [1]. Different strategies are applied in certain probability in every training iteration for more randomness.

Adversarial Attack We make adversarial attack on text embeddings for better generalization. Adversarial attack means generating adversarial examples that significantly increase the loss incurred by neural networks. Adversarial examples are usually small perturbations to the input. In this way, not only the robustness of the model is improved, but also the generalization performance is enhanced. We apply adversarial attack to the text transformer following Fast Gradient Method (FGM) [5]. The adversarial perturbation is

$$r_{adv} = \epsilon \cdot g/||g||_2 \; where \; g = \nabla_x L(y \mid x; \theta) \qquad (1)$$

where x is the input and θ stands for the parameters of the text embedding layers. We set ϵ as 1.0 in our experiments.

Imporved Contrastive Loss Function Considering that there are probably the same attributes in one batch of training examples since vehicle attributes are limited, we make some adaptations to the original CLIP contrastive loss function. Given a batch of N (image, text) pairs, CLIP learns a multimodal embedding space by jointly training an image encoder and text encoder to maximize the cosine similarity and text embeddings of the N real pairs while minimizing the cosine similarity of the left $N^2 - N$ pairs. Next, it optimizes a symmetric cross-entropy loss over these similarity scores. However, different from CLIP with 400 million text-image pairs of plenty of categories, there are also matching ones among $N^2 - N$ pairs during our task. Therefore, we replace cross-entropy loss with Kullback–Leibler(KL) divergence loss for more precise matches in training [8]:

$$L = \frac{1}{2}\mathbb{E}_{(x,y)\sim D}\left[KL(p^{x2y}(x), g^{x2y}(x)) + KL(p^{y2x}(y), g^{y2x}(y))\right] \qquad (2)$$

where $g^{x2y}(x)$ and $g^{y2x}(y)$ indicate the ground-truth similarity scores, $p^{x2y}(x)$ and $p^{y2x}(y)$ stand for the predictions. D means the entire training dataset.

3. Experiments

In this section, we conduct comprehensive experiments on the dataset. And the comparisons of mean average performance are presented to verify the effectiveness of the proposed method. The results are in Table 3.

3.1. Training Tricks

Model Scale The configuration arguments of ViT become bigger as the scale increases. From ViT-Base to ViT-Large, the number of stacked encoder blocks doubles. Moreover, the dimension of embeddings layers and the number of attention heads grows. The capability of the model has been significantly improved with the increase of parameters. As a result, there is a mAP gain of about 3.5 points from ViT-Base to ViT-Large.

Weights Exponential Moving Average EMA is utilized in our solution to average the parameters of the model for robustness. We use the momenta parameter of 0.999 during training and save the EMA weights at the end of training.

Stratified K-Folds Cross-validation To avoid selection bias and overfitting, we conduct Stratified K-Folds cross-validation on the dataset. Specifically, we randomly split the entire dataset into 14 folds while keeping the account of each class the same in every fold. Due to training overhead, we only select 5 of 14 folds for experiments. During inference, we fuse the outputs of these models for stable predictions.

3.2. Experimental Settings

We explore different methods in several aspects including network scale, adversarial attack, and contrastive loss function. The performance of Vision Transformer (ViT) variants is evaluated on the validation set. We train ViT-Base without freezing layers. Except for the 6 layers near the output, the parameters of other layers are fixed during training since the weights of the original CLIP are more generative. And ViT-Large is adopted as the backbone of our submission.

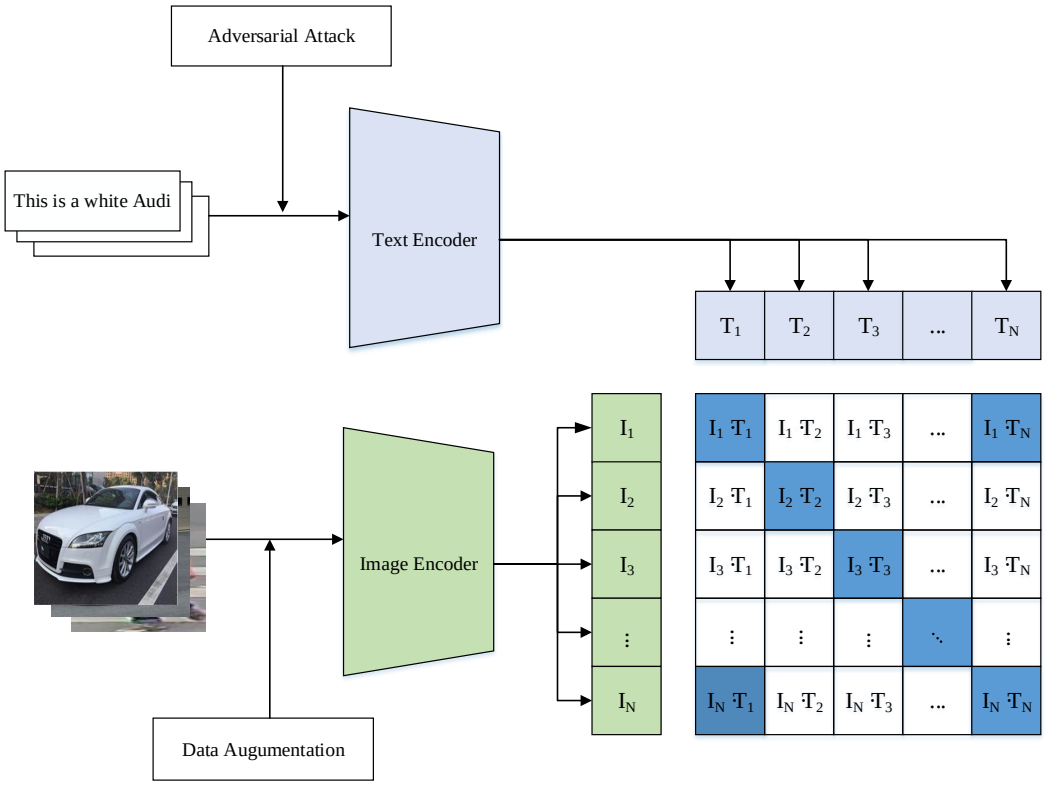

Figure 2. The overall network architecture. The network consists of an image encoder, a text encoder, and the embedding projection module. In the image encoder, we use strong data augmentation. In the text encoder, we make adversarial attack to text embeddings. And in the embedding projection module, we promote the positive samples assignment strategy.

Table 3. Results of methods on Leaderboard A/B. We make comparisons between different model scales, tricks, and fusion strategies. When ViT-Large, EMA, AutoAug, FGM, and *add sim* are composed, we obtain the best score in Leaderboard A.

Scale		Trick			Fusion			Leaderboard	
ViT-Base	ViT-Large	EMA	AutoAug	FGM	cat feat	add feat	add sim	A	B
✓		✓						69.649	
	✓	✓						73.114	
	✓	✓	✓					74.633	
	✓	✓		✓				74.837	
	✓	✓	✓	✓	✓			76.264	
	✓	✓	✓	✓		✓		75.951	
	✓	✓	✓	✓			✓	**76.268**	**68.5**

All the experiments are conducted by using of PaddlePaddle [4], which is an open-source deep learning platform developed by Baidu, Inc. And the code is based on UFO [9]. We run experiments on 8 NVIDIA A40 GPUs with batch size of 1024. We adopt the AdamW optimizer as well as a CosineAnnealingLR scheduler with base learning rate of 5e-4. We train the model for 20 epochs, the ratio of WarmUp is 0.1, and the initial learning rate is set to 5e-5. The input image size is 224*224 since larger size input brings no gain in mean average precision in experiments. The model is evaluated every epoch, and only the best one is kept.

3.3. Testing Tricks

During inference, we also use test time augmentation and different model fusion methods to improve the precision and stability of predictions. Quantitative results are shown in Fig. 3.

Test Time Augmentation In experiments, we find that simply averaging image embeddings of the original image and the horizontal flip image itself can improve the mAP. Besides, we also try to use the average of FiveCrop from the original image. However, it does not generate higher mAP

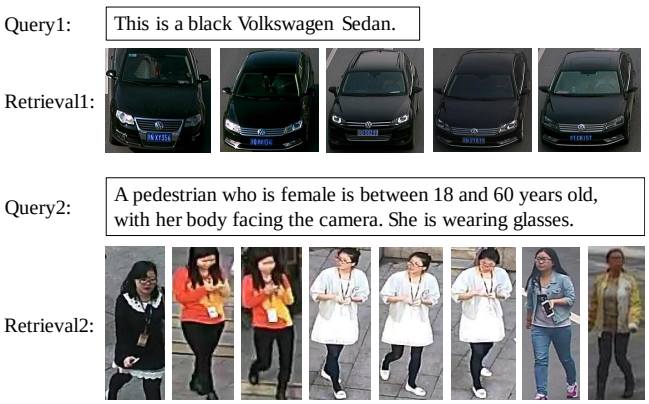

Figure 3. The quantitative results of retrieval. The results of vehicles are better than pedestrians for the attribute features are relatively simpler than pedestrians.

than the former method.

Model Fusion Based on the five models trained by the cross-validation method, we explore different model fusion strategies including *add sim*, *add feat*, and *cat feat*. *Add sim* means adding output similarity matrices of five models, *add feat* means averaging the image embeddings generated by five models, and *add feat* means concatenating image embeddings in the last dimension. In our experiments, all three methods can slightly improve the mAP, and *add sim* generates the highest score of 76.268 in Leaderboard A as well as 68.5 in Leaderboard B.

4. Conclusion

The proposed method for Cross-Modal Image Retrieval Track of 1st foundation model challenge is illustrated in this report in detail. We explore data augmentation, model structure, and training tricks through comprehensive experiments. Strong data augmentations are applied first to the training pipeline, then proper model scale and input image size are studied. We optimize the contrastive loss function and make adversarial attack during training. After that, SKF, EMA, TTA, and model fusion are used for better model performance and robustness. By assembling these strategies, we achieve a competitive performance in leaderboard B, which demonstrates the effectiveness of our solution.

In future work, we are going to reflect on methods that work theoretically, nonetheless fail to improve the mAP. For example, we made an attempt to complete vehicle attributes by selecting the attributes shared the highest similarity with the image through prompting. We also try to introduce the moving-averaged encoder from MoCo [2] into our solution. We hold that model fusion between different network architectures like Transformer and ConvNext [3] can achieve better performance. We hope our solution can arouse insight and contribute to the development of the foundation model society.

References

[1] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018. 2

[2] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 4

[3] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022. 4

[4] Yanjun Ma, Dianhai Yu, Tian Wu, and Haifeng Wang. Paddlepaddle: An open-source deep learning platform from industrial practice. *Frontiers of Data and Domputing*, 1(1):105–115, 2019. 3

[5] Takeru Miyato, Andrew M Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*, 2016. 2

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[8] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition. *arXiv preprint arXiv:2109.08472*, 2021. 2

[9] Teng Xi, Yifan Sun, Deli Yu, Bi Li, Nan Peng, Gang Zhang, Xinyu Zhang, Zhigang Wang, Jinwen Chen, Jian Wang, et al. Ufo: Unified feature optimization. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pages 472–488. Springer, 2022. 3