Direct Projection of Visual Features into Linguistic Embeddings for Domain-Adapted Multimodal Learning

Jordan Kralev
Department of Systems and Control, Technical University of Sofia, Bulgaria
Institute for Bulgarian Language, Bulgarian Academy of Scienses
jkralev@tu-sofia.bg, jordan@dcl.bas.bg

Abstract

This paper introduces a novel multimodal learning framework that directly projects visual features into linguistic embeddings, enabling robust, domain-adapted object recognition across diverse categories. Leveraging state-of-the-art visual (Detectron2 Mask R-CNN) and language (TinyLlama) models, the architecture incorporates a trainable projection layer that maps convolutional visual features into the embedding space of a large language model. This approach addresses the challenge of bridging high-dimensional, region-based visual representations with sequential, token-based language embeddings using minimal fine-tuning. The model is evaluated on the specialised Multilingual Image Corpus (MIC21), a rich dataset containing over 21,000 images and more than 700 categories, extending beyond standard benchmarks and annotated with multilingual captions. Experimental results demonstrate the effectiveness of this system for multimodal tasks, including image captioning and visual question answering. The design supports scalable adaptation to new domains and tasks, offering a flexible solution for unified visual–linguistic representation learning.

Keywords: multimodal learning, visual-linguistic embeddings,domain adaptation,object recognition

1 Introduction

Despite significant advances in deep learning models for processing complex signals – visual, audio, and linguistic – large models still struggle to exploit complexity by fusing domain-specific data. Applications for multimodal models that process visual and textual data are evident in medicine, autonomous systems, education, and manufacturing, among other domains. Tasks such as image captioning and visual question answering depend on both image and textual data.

The current state-of-the-art framework for visual large language models (LLMs) is CLIP (Contrastive Language–Image Pre-training). Introduced by OpenAI, it connects visual and linguistic information using contrastive objectives. Its design comprises two parallel encoders trained jointly: the text encoder converts a sentence into a fixed-length embedding reflecting its semantic meaning, while the image encoder transforms an input image into a vector summarising its visual content. Through contrastive learning, the model aligns matching image–text pairs in a shared embedding space while separating unrelated ones, enabling it to assess how closely a caption corresponds to a given image. OpenAI CLIP models are trained on WebImage-Text, a dataset of 400k image/caption pairs. The total word count of this dataset is comparable to that of the WebText dataset, which contains about 40 terabytes of text data and was used to train GPT-2 (Radford et al., 2021). Image–text pairs are usually only loosely correlated, as a caption can match multiple images in addition to the ground truth (Cheng et al., 2021). This is also reflected in the discriminative ability and transfer performance of the visual encoder pre-trained with a contrastive objective (Yang et al., 2022).

There are visual models for self-supervised learning (SSL) that have been pre-trained with non-contrastive objectives (Caron et al., 2021). Beyond the contrastive paradigm, some visual SSL models succeed in mitigating dependence on negative samples (Schiappa et al., 2023). With various refinements to avoid collapsing solutions, these works optimise the affinity of augmented representations alone and are categorised as non-contrastive frameworks. For example, to avoid model collapse, asymmetric architectures (Chen and He, 2021), dimensional decorrelation (Bardes et al., 2022), and clustering (Assran et al., 2022; Schiappa et al., 2023) are employed.

Aligning the high-dimensional, region-level visual embeddings produced by modern object detectors such as Detectron2 or YOLO with the sequential, tokenised representations of language models such as Llama presents a significant technical challenge. Simple strategies such as direct concatenation or shallow fusion are expected to yield poor performance because the modalities operate in incompatible feature spaces and lack meaningful cross-modal interactions. While state-of-the-art frameworks such as CLIP rely on extensive training of a combined visual and textual embedding matrix, the present work adopts a more system-level approach, seeking to establish a connection between robust pre-trained visual and linguistic models with minimal fine-tuning of the connecting layers. The goal of this article is to develop a general and adaptable framework that enables object recognition to transfer across domains with custom categories through a learnable projection layer, unifying visual and linguistic embeddings to support robust multimodal training for diverse downstream tasks.

The study examines possible interconnections between the Detectron2 Mask R-CNN visual model and the TinyLlama large language model (LLM), linked through a trainable projection layer from the convolutional feature space into LLM embeddings. The performance of the architecture is demonstrated on a specialised multi-domain dataset called the Multilingual Image Corpus (MIC21). The original purpose of the MIC21 dataset was to expand the 80 COCO categories, commonly used for image processing models, into an extensive ontology featuring over 700 classes, organised into 130 thematic subdomains covering sport, transport, arts, and security. Currently, the dataset contains around 21,000 images with more than 200,000 object detections. It was recently extended with textual captions and descriptions for each image. Images were collected from diverse public sources and annotated based on a custom ontology, which is linked to WordNet and translated into 25 languages, providing rich semantic detail and multilingual support. While the text-generation model developed in the paper is English-only, the design could be extended for multilingual support. A fine-tuning approach to Detectron2 for domain-specific data (outlined in previous work by the authors) can further improve the capacity for knowledge transfer between models through the projection layer.

The paper is organised as follows. Section 2 describes the model architecture. Section 3 presents the training strategy, and Section 4 discusses the experimental results and provides concluding remarks.

2 Model Architecture

The proposed model is interconnection between backbone components of two heterogeneous deep learning model. The first component is derivation from Facebook Detectron2 visual model of it is a Mask R-CNN with X-101-32x8d-FPN configuration. For a given RGB input image $I \in \mathbf{R}^{H \times W \times 3}$ with height H and width W pixels the backbone component of the FPN extracts a tensor

$$x_i \in \mathbf{R}^{C \times H_F \times W_F}, \tag{1}$$

where $C = 256$ is the filter dimension of the final convolutional layer, $H_F = 16$ and $W_F = 16$ are the dimensions of the convolution map. The feature tensor x_i is then flattened into

$$x_f \in \mathbf{R}^{C \times (H_F W_F)} \tag{2}$$

The feature matrix is then normalized and projected to the embedding dimension d

$$v = W_p F_{drop}(F_{norm}(x_f)) + b_p \tag{3}$$

through the matrix $W_p \in \mathbf{R}^{d \times C}$ and bias vector $b_p \in \mathbf{R}^d$ and $v \in \mathbf{R}^{256 \times d}$, $d = 2048$ for the TinyLlama model. Here W_p and b_p are the trainable parameters of the proposed projection layer. The normalization layer F_{norm} applies a conditioning transform

$$x_{fn} = \gamma \frac{x_f - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta \tag{4}$$

where μ and σ are mean and standard deviation of the input vector, while γ and β are trainable parameters. Also during training a dropout filter F_{drop} is applied.

The input to the language processing component of the architecture is a token sequence $t = (t_1, t_2, \ldots, t_T)$. In the language model each token is transformed to an embedding vector to get

$$e = (e_1, e_2, \ldots, e_T) \in \mathbf{R}^{T \times d} \tag{5}$$

where e_i is the embedding of token t_i. Then the projections from the visual convolutional layer are

injected into the prompt so the input to the transformer layer is a concatenation of projected visual tokens and text embeddings

$$z_0 = (e_{pre}, v, e_{post}) \in \mathbf{R}^{(256+T)\times d} \tag{6}$$

The position for injecting visual features into the prompt is chosen to match the model's instruction template and to provide sufficient context to clarify the task at hand. As is well known, the core of sequence generation is a stack of transformer decoder layers. The decoder consists of L layers, each applying multi-head self-attention.

$$a_l = f_{smax}\left(\frac{QK^T}{\sqrt{d}} + M\right)V \tag{7}$$

where queries Q, keys K and values V are all linear projections of z_{l-1}, and $M \in \mathbf{R}^{(256+T)\times(256+T)}$ is causal mask defined with

$$M = \begin{pmatrix} 0_{256\times256} & 0_{256\times T} \\ 0_{T\times256} & U_{T\times T} \end{pmatrix} \tag{8}$$

where U is upper triangular matrix with entries

$$u_{i,j} = \begin{cases} 0, & i \geq j \\ -\infty, & i < j \end{cases} \tag{9}$$

The decoder operates in an autoregressive manner: it starts with an initial empty token embedding and, at each step, generates the next token embedding conditioned on the previously generated sequence and the visual context. This process is repeated up to the maximum sequence length, allowing the model to generate a sequence of tokens that describe the input image. The model output is projected into token indices in a deterministic manner as

$$y = argmax_i(z_i), \quad (z_i) \in \mathbf{R}^d \tag{10}$$

3 Training Strategy

Both the visual and linguistic components of the model are frozen during training, focusing the learning capacity on the projection and normalisation layers. Consider a general sequence of tokens

$$w = (w_1, w_2, \ldots, w_T) \tag{11}$$

where $w_t \in 1 \ldots m$ is the index of a token in the vocabulary. The model output is generating a conditional distribution over sequences of tokens for a given input image I defined autoregressively as

$$p(w, I, \theta) = \prod_{t=1}^{T-1} p(w_{t+1}|w_{1:t}, I, \theta) \tag{12}$$

where θ is the vector of tunable parameters. The training goal is to maximize the log-likelihood of the image-caption pairs from the training data set. This requires minimization of the negative log-likelihood

$$\begin{aligned} \mathcal{L} &= -\log p(w, I, \theta) \\ &= -\sum_{t=1}^{T-1} \log p(w_{t+1}|w_1, \ldots, w_t, I, \theta) \end{aligned} \tag{13}$$

For each position $t = 1 \ldots T$, let $z_t \in \mathbf{R}^m$ be the model's output (so called logits) for the next generated token over vocabulary of size m. After application of softmax operation to convert logits to probabilities we can get an explicit expression for $p(w, I, \theta)$, i.e.

$$p(w_{t+1} = i|w_1, \ldots, w_t) = \frac{e^{z_{t,i}}}{\sum_{j=1}^{m} e^{z_{t,j}}} \tag{14}$$

where $z_{t,j}$ is element at position j from the vector z_t. In the proposed architecture *autoregressive generation loss* is employed where past sequence $w_{1:t}$ is obtained from previously generated tokens, i.e.

$$w_t \sim p(w_t|w_{1:t-1}, I, \theta) \tag{15}$$

In this case, each $\mathcal{L}_t$ depends on all previous predictions through the recursive generation. The gradient calculation $\partial\mathcal{L}/\partial\theta$ will require backpropagation through t unrolled steps. Such an approach is not advisable for models with many layers, as it can lead to vanishing or exploding gradients, along with high memory usage. However, in the proposed model architecture, the only trainable parameters are the projection layer weights W_p and biases b_p. The projection layer parameters contribute to the loss gradient as

$$\frac{\partial\mathcal{L}}{\partial W_p} = \sum_{t=1}^{T}\left(\frac{\partial\mathcal{L}_t}{\partial z_t}W_e\right)\frac{\partial y_t}{\partial v}\frac{\partial v}{\partial W_p} \tag{16}$$

4 Results

The tuning process begins with the selection of pre-trained versions of the visual and linguistic model. For the visual features encodings Detectron2 model for COCO instance segmentation task is used with a structure of mask RCNN, signature X_101_32x8d_FPN_3x. The linguistic counterpart for the experiment is TinyLlama-1.1B-Chat-v1.0. The hardware configuration for training and evaluation consists of 2 x NVIDIA GeForce GTX 1080

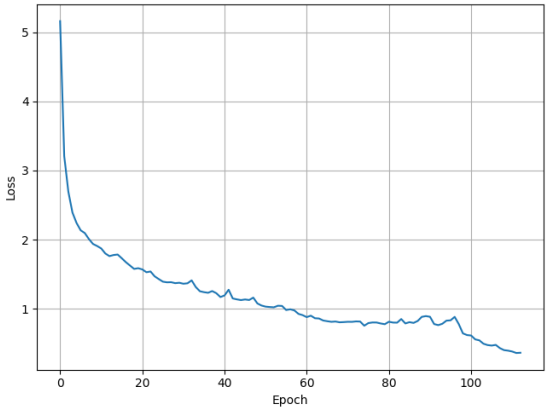

Figure 1: Average loss function during training.

Figure 2: Example input image along with detected objects from Detectron2.

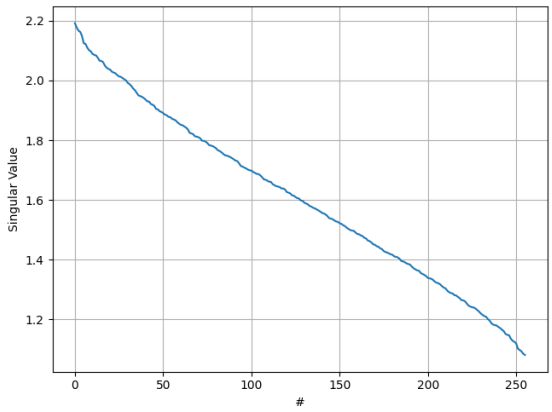

Figure 3: Singular values of the projection matrix before training.

and 1 x NVIDIA GeForce RTX 3070. The training is performed for 115 epochs for a domain of 'cricket' from MIC21 dataset with a batch size of 7 and 40 for max generation sequence. The batch size and sequence length are tuned to optimize for the architecture of 3 GPUs. The layers and weights of the Llama models are distributed between the 3 cards to occupy upto 3GB of the both GTX cards and upto 5GB of the RTX card. The Detectron model is entirely loaded into RTX card. The caching is enabled to accelerate autoregressive generation.

The learning rate for training is set at 0.0001 and using Adam optimizer. The average loss per epoch can be seen at Fig. 1. The loss function show strong initial decrease, followed by slower but constant slope and final drop after 100th epoch, where actual human quality captions begin to appear. Example image and respective detections from Detectron2 model are presented into Fig. 2. As it can be seen images contain diverse features and colours. The presented detected objects are obtained from the consecutive Detectron2 layers, which are not projected to LLM, however they could be integrated into future research iterations.

The generation prompt is:

```
<|system|>
Generate title and description for the
provided image. The image features are:
<v1><v2>...<v256></s>
<|user|>
Generate a title:</s>
<|assistant|>
```

As shown, the prompt contains a short system message followed by a user instruction. The visual features are injected at the end of the system message as a sequence of token embeddings obtained from the projection layer. In this way, each image adds 256 tokens to the prompt, which guide subsequent output generation.

The structure of the projection matrix $W_b \in \mathbf{R}^{256\times2048}$ is examined through singular value decomposition to assess the magnitude of training. The singular values of W_b before training are shown in Fig. 3. This pre-training spectrum indicates that the matrix is well conditioned and capable of representing high-dimensional data without severe rank deficiency, providing a solid foundation for effective adaptation during subsequent learning.

The singular values of W_b after training are shown in Fig. 4. Following training, the singular value spectrum of the projection matrix shows a marked change, with a pronounced concentration of energy in the leading singular value, which now exceeds 3.5, and a steeper drop-off among the subsequent components. This pattern suggests that training has polarised the directions in the matrix: a few dominant modes now carry significantly more

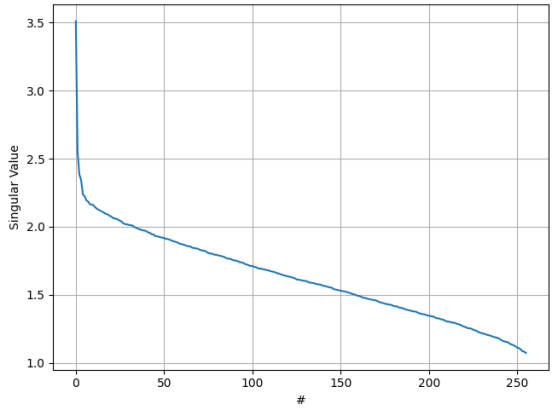

Figure 4: Singular values of the projection matrix after training.

Metric	Epoch 115	Epoch 0
BLEU	0.7	0.0
ROUGE 1	0.84	0.005
ROUGE 2	0.74	0.0
ROUGE L	0.83	0.005
ROUGE L Sum	0.83	0.005
METEOR	0.81	0.004
Precision	0.93	0.46
Recall	0.92	0.52
F1 Score	0.92	0.49
WER	0.23	1.35

Table 1: Model performance comparison for cricket domain.

information, while the remaining dimensions contribute less to the representation. Such behaviour is typical of learned projection layers, in which the network identifies and amplifies key directions to bridge visual and linguistic feature spaces more effectively. The compression of singular values beyond the first few may reflect emergent sparsity or the prioritisation of task-relevant structure, which can improve robustness and generalisation in downstream multimodal learning.

Table 1 presents an evaluation of model performance in the cricket domain before and after domain-specific training on the MIC21 dataset. The metrics shown – BLEU, ROUGE, METEOR, Precision, Recall, F1 Score, and Word Error Rate (WER) - capture both the semantic and structural accuracy of generated captions or outputs. At Epoch 0, the model exhibits near-zero scores in BLEU, ROUGE, and METEOR, along with moderate Precision, Recall, and F1 Score, while suffering a high WER of 1.35. These baselines illustrate that the untrained model is unable to accurately capture the domain-specific content or structure required for high-quality generation.

By Epoch 115, every metric shows dramatic improvement. BLEU and ROUGE scores rise to competitive levels, indicating that the model now generates text with strong fidelity to reference sentences. METEOR also improves substantially, confirming better semantic matching. Precision, Recall, and F1 Score all exceed 0.9, highlighting robust classification and retrieval capabilities, while WER drops markedly to 0.23, indicating a substantial reduction in transcription errors. Altogether, these advances demonstrate that the domain-adapted multimodal system is capable of highly accurate and reliable output when sufficiently trained on a targeted domain dataset.

5 Conclusion

The paper presents a robust and flexible approach to multimodal learning by directly projecting high-dimensional visual features into linguistic embedding spaces through a trainable projection layer. By leveraging strong pre-trained models for both vision (Detectron2 Mask R-CNN) and language (TinyLlama), the framework addresses the challenge of cross-modal fusion with minimal fine-tuning, enabling accurate, domain-adapted object recognition, captioning, and question answering. Experiments on the comprehensive MIC21 dataset – covering over 21,000 images and 700 categories – show substantial gains in all key performance metrics after targeted training, with notable improvements in BLEU, ROUGE, METEOR, precision, recall, and word error rate.

Analysis of the singular value spectra before and after training further validates the projection layer's ability to extract and prioritise meaningful cross-modal directions. This work lays a foundation for scalable domain adaptation in unified visual–linguistic models and opens new perspectives for extending the architecture to broader domains and multilingual tasks.

Acknowledgments

The present study is carried out within the project Infrastructure for Fine-tuning Pre-trained Large Language Models, Grant Agreement No. ПВУ – 55 from 12.12.2024 /BG-RRP-2.017-0030-C01/.

References

Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Mike Rabbat, and Nicolas Ballas. 2022. Masked siamese networks for label-efficient learning. In *ECCV (31)*, volume 13691 of *Lecture Notes in Computer Science*, pages 456–473. Springer.

Adrien Bardes, Jean Ponce, and Yann Lecun. 2022. VICReg: Variance-Invariance-Covariance Regularization For Self-Supervised Learning. In *ICLR 2022 - International Conference on Learning Representations*, Online, United States.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 15750–15758. Computer Vision Foundation / IEEE.

Ruizhe Cheng, Bichen Wu, Peizhao Zhang, Peter Vajda, and Joseph E Gonzalez. 2021. Data-efficient language-supervised zero-shot learning with self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3119–3124.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020.

Madeline C. Schiappa, Yogesh S. Rawat, and Mubarak Shah. 2023. Self-supervised learning for videos: A survey. *ACM Comput. Surv.*, 55(13s).

Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, and Jianfeng Gao. 2022. Unified contrastive learning in image-text-label space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19163–19173.