# Fusion of Object-Centric and Linguistic Features for Domain-Adapted Multimodal Learning

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

## Abstract

Modern multimodal systems often struggle to link domain-specific visual content with textual descriptions, especially when object recognition is limited to general categories (e.g. COCO classes) and lacks customised adaptation to language models. In this paper, we present a novel framework that integrates a domain-specific adapted Detectron2 model into predefined models via a trainable projection layer, enabling precise crossmodal adaptation for specialised domains. Our approach extends Detectron2's recognition capabilities to new categories by fine-tuning on multi-domain datasets, while a lightweight linear projection layer maps region-based visual features to the model's embedding space without completely retraining the model. We evaluated the framework for domain-specific image captioning. The presented approach provides a scalable design for combining domain-specific visual recognition with language inference, with applications in domains that require fine-grained multimodal understanding. The results show rapid model convergence and adaptation to specialized domains. The system achieves competitive BLEU scores on caption generation, though slightly below Gemma 3 baseline.

## 1 Introduction

Despite remarkable advances in computer vision and natural language processing, most modern models are still largely unimodal and are characterised by either visual understanding (e.g. object recognition) or language understanding (e.g. text generation), but not both. In real-world applications - such as image captioning, visual question answering and multimodal retrieval - there is a growing need for models that can interpret both images and text. Existing multimodal approaches often rely on generic visual features extracted from object detectors trained on standard datasets such as COCO (Lin et al., 2014), which limits their applicability to specialised domains where object categories and visual semantics differ significantly.

Furthermore, it is a major challenge to match the high-dimensional, region-based features generated by advanced object detectors such as Detectron2 with the sequential, token-based representations used by language models. Naive concatenation or superficial fusion of these features often leads to suboptimal performance due to mismatching feature spaces and insufficient cross-modal interaction. This motivates the goals of our work - to propose a robust and flexible system that adapts object recognition to new domains with user-defined categories, effectively bridges the gap between visual and linguistic representations using a trainable projection layer, and enables seamless end-to-end learning for a wide range of multimodal tasks.

This study tackles these issues by introducing a scalable system that combines a domain-specific Detectron2 model with BERT, linked through a trainable projection layer, showcasing its performance on specialized multi-domain datasets. The fine-tuning approach for Detectron2, as outlined in a previous authors' work, is integrated with the creation and utilization of the Multilingual Image Corpus (MIC21). MIC21 expands the conventional 80 COCO categories into an extensive ontology featuring over 700 classes, categorized into 130 thematic subdomains including Sport, Transport, Arts, and Security. The dataset comprises more than 21,000 images and 200,000 annotations. Annotation involved a hybrid process of automated segmentation and classification followed by thorough manual refinement. 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



*Proceedings of Recent Advances in Natural Language Processing*,pages 587–594
Varna, Sep 8–10, 2025
https://doi.org/10.26615/978-954-452-098-4-069

is English-only the design could be extended for multilingual support.

The primary contribution of this paper is the introduction of a scalable and modular multimodal framework that tightly integrates a domain-adapted Detectron2 object detection model with a BERT-based language model through a learnable projection layer, enabling precise cross-modal adaptation for specialized domains. The system is designed to map high-dimensional, region-based visual features into the linguistic embedding space efficiently, allowing for seamless end-to-end learning without retraining large pre-trained components. This architecture is evaluated on domain-specific image captioning tasks.

The organisation of the paper is as follows. The Section 2 reviews recent advances in multimodal learning. Section 3 overview the dataset creation, annotation workflows, and the fine-tuning of Detectron2 models. Section 4 is devoted to the model architecture. Sections 5 and 6 present experimental results, discussion and concluding remarks.

## 2 Related Work

Recent advances in the field of multimodal learning have focussed on the combination of computer vision and natural language processing.

CLIP (Contrastive Language-Image Pre-training)[1] is a multimodal AI framework developed by OpenAI that learns to combine images and natural language through contrastive learning. The CLIP approach trains two models in a contrastive way (Radford et al., 2021). The text encoder processes an input sentence and transforms it into a fixed-dimensional vector that encapsulates its semantic meaning. Conversely, the image encoder takes an input picture and similarly produces a corresponding vector that captures its visual content.

The core idea of CLIP is to train two separate encoders – an image encoder (often a vision transformer or CNN) for images and a transformer-based language model for text – on a huge dataset of image-text pairs from the Internet. Each encoder transforms its input (an image or a text description) into a high-dimensional vector in a shared embedding space. During training, CLIP uses a contrastive loss: it encourages the embeddings of matching pairs (image, text) to be close to each other, while it pushes the embeddings of non-matching pairs apart. This procedure allows the model to learn rich, general visual and semantic concepts directly from natural language supervision instead of relying on manually labelled datasets.

WebImageText, a dataset of 400 pairs of images and their captions from the Internet, was used to train the OpenAI CLIP models. 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 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 or transferring performance of the visual encoder pre-trained with the contrastive objective (Yang et al., 2022).

On the other hand, there are visual models for self-supervised learning (SSL) that have been pre-trained with the non-contrastive objective (Caron et al., 2021). Beyond the contrastive paradigm, some visual self-supervised learning (SSL) models succeed in mitigating dependency on negative samples (Schiappa et al., 2023). With various refinements to avoid collapsing solutions, these works can optimise the affinity of the augmented representations alone and are categorised as a non-contrastive framework. 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 used.

In summary, let state clearly state what is existing and what is new in the proposed approach. Existing datasets and models are: Detectron2 and Yolact Models; BERT Language Model; COCO Image Dataset. The novel contributions are: MIC21 Multilingual Dataset; Fine-Tuned Detectron2 Model per Domain; MIC21Summarizer Model Architecture; Training Strategy.

## 3 Dataset Preparation

The fine-tuning process begins with the use of pre-trained models such as Yolact and Detectron2, which are first trained on the COCO dataset. These models are used to generate preliminary object boundaries and class labels for the MIC21 images. The automatically generated annotations are then imported into the COCO annotator tool, where human annotators correct the segmentation masks and

[1] https://openai.com/index/clip/

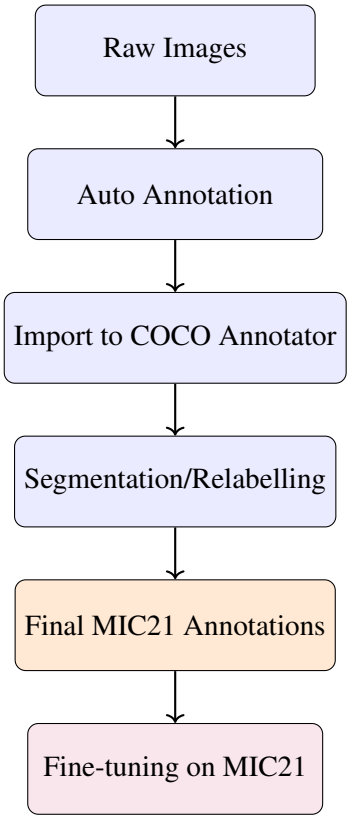



Figure 1: Workflow for dataset creation and annotation in MIC21.

re-label objects according to the MIC21 ontology. This process is further optimised by custom scripts that automate dataset management tasks, such as relabelling, merging and exporting annotated data (Figure 1).

Once the annotated dataset is prepared, the Detectron2 framework is used for model training. The architecture typically consists of a ResNet-based backbone with a Feature Pyramid Network (FPN), a Region Proposal Network (RPN), and an ROI head for classification and mask prediction. During fine-tuning, the backbone and RPN layers are often frozen to retain the general visual features learned from COCO, while the ROI head is retrained to adapt to the new, domain-specific classes of MIC21.

The models are assessed using COCO-style metrics, including precision, recall, and average precision (AP) across a range of intersection-over-union (IoU) thresholds. The FiftyOne framework is used for visualization and comparison of model predictions with ground truth annotations (Figure 2).

## 4 Model Architecture

The developed model, MIC21Summarizer, is an advanced multimodal neural network architecture designed to generate textual descriptions from images by integrating state-of-the-art computer vision and natural language processing components (Figure 3). The architecture begins with an image feature extraction backbone based on Detectron2's Mask R-CNN with an X-101-32x8d-FPN configuration. This model is pre-trained on the COCO dataset for instance segmentation. The image is preprocessed and passed through the backbone, and features are extracted from the 'p6' feature map.

Let $I \in \mathbf{R}^{H\times W\times 3}$ be the input image. The Detectron2 backbone extracts a feature tensor

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

where in our model $C = 256$, $H_F = 16$ and $W_F = 16$. For this purpose a $16 \times 16$ pooling layer is applied over Detectron2 'p6' features which are with variable size dependent on image size. The feature tensor $F$ is reshaped into a matrix

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

The feature matrix is normalized and projected to the embedding dimension $d$ (here $d$ = 1024 for bert-cased-large)

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

where $W_p \in \mathbf{R}^{d\times C}$, $b_p \in \mathbf{R}^d$ and $v \in \mathbf{R}^{256\times d}$.

For the language component, the model leverages the BERT architecture. It initializes a BERT tokenizer and the corresponding embedding layer from the "bert-large-cased" model. For a token sequence $t = (t_1, t_2, \ldots, t_T)$, the BERT embedding layer produces

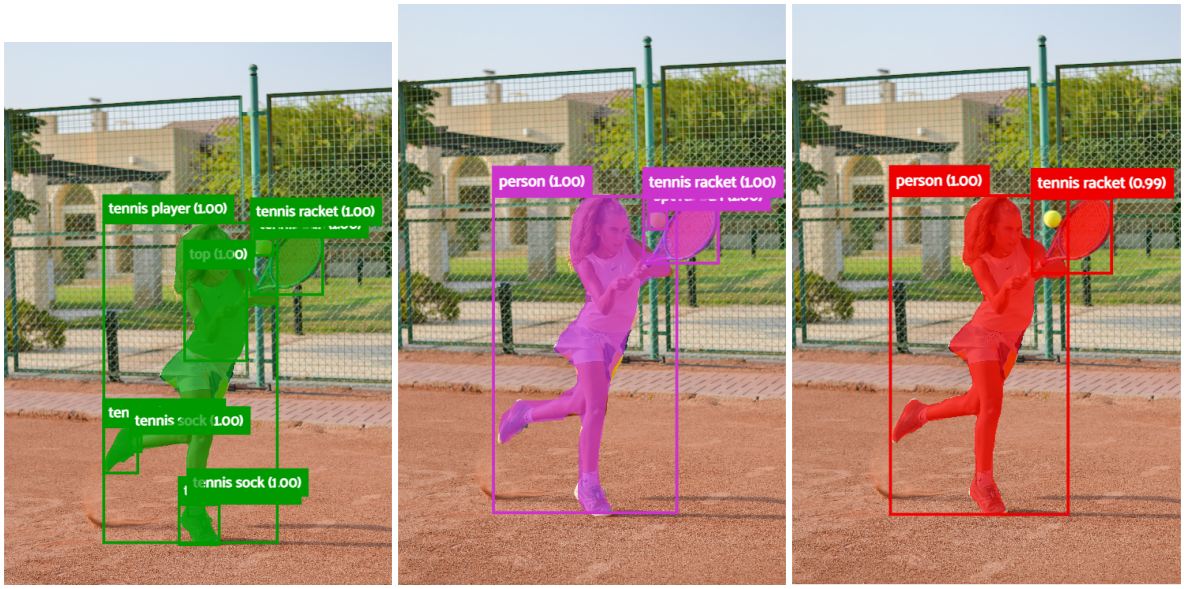
$$e = (e_1, e_2, \ldots, e_T) \in \mathbf{R}^{T\times d} \tag{4}$$

where $e_i$ is the embedding of token $t_i$. The input to the decoder is the concatenation of projected visual tokens and text embeddings

$$z_0 = (v, e) \in \mathbf{R}^{(256+T)\times d} \tag{5}$$

The core of the sequence generation is a stack of three Transformer Decoder layers, each with eight attention heads and a model dimension of $d = 1024$. The decoder consists of $L$ layers (here $L$=3), each applying multi-head self-attention

$$a_l = f_{smax}\left(\frac{QK^T}{\sqrt{d}} + M\right)V \tag{6}$$




(a) Human (b) Detectron2 (c) YOLACT

Figure 2: Example of ground-truth annotation and annotation results using Detectron2 and YOLACT

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

$$M = \begin{pmatrix} 0_{256\times256} & 0_{256\times T} \\ 0_{T\times256} & U_{T\times T} \end{pmatrix} \tag{7}$$

where $U$ is upper triangular matrix with entries

$$u_{i,j} = \begin{cases} 0, & i \geq j \\ -\infty, & i < j \end{cases} \tag{8}$$

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 for up to 128 steps, allowing the model to generate a sequence of tokens that describe the input image.

The output from the decoder (only last T positions) is projected onto the vocabulary

$$y = W_e^T z_{l,256:T} \in \mathbf{R}^{T\times m} \tag{9}$$

where $W_e$ are the frozen BERT word embeddings, and $m$ is the vocabulary size. The word embedding weights are reused in the output projection layer, ensuring that the output tokens generated by the decoder are aligned with the BERT vocabulary and tokenization. Token predictions are made via an argmax over the logits.

The aim of the model is to tightly couple high-level visual understanding with transformer architecture, enabling powerful language modeling capabilities. The design emphasizes modularity and efficiency by freezing large pre-trained components and focusing training on the crucial projection and decoding layers.

### 4.1 Training Strategy

Both the image backbone and the BERT embedding layers are frozen during training, focusing learning capacity on the projection and decoding components. To perform cross-entropy training with one token ahead prediction in autoregressive models we optimize the model to predict the next token in a sequence given the previous tokens, known as teacher forcing.

Given a sequence of tokens

$$w = (w_1, w_2, \ldots, w_T) \tag{10}$$

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

$$p(w, I, \theta) = \prod_{t=1}^{T-1} p(w_{t+1}|w_{1:t}, I, \theta) \tag{11}$$

where $\theta$ is the vector of tunable parameters.

The training goal is to maximize the log-likelihood of the image-sequence pairs from the

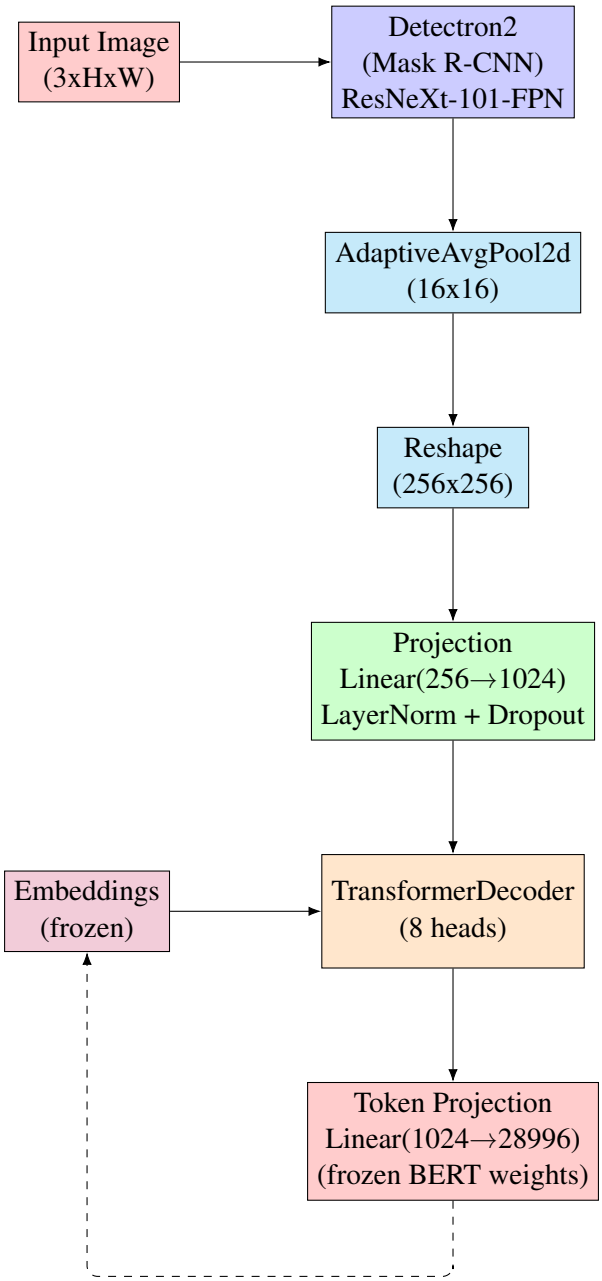



Figure 3: MIC21Summarizer Architecture Overview

training data set. This translates to minimizing the negative log-likelihood

$$\mathcal{L} = -\log p(w, I, \theta) = \\ -\sum_{t=1}^{T-1} \log p(w_{t+1}|w_1, \ldots, w_t, I, \theta) \quad (12)$$

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.

$$p(w_{t+1} = i|w_1, \ldots, w_t) = \frac{e^{z_{t,i}}}{\sum_{j=1}^{m} e^{z_{t,j}}} \quad (13)$$

where $z_{t,j}$ is element at position $j$ from the vector $z_t$.

Two training modes are possible in this framework. One is *autoregressive generation loss* where past sequence $w_{1:t}$ is obtained from previously generated tokens, i.e.

$$w_t \sim p(w_t|w_{1:t-1}, I, \theta) \quad (14)$$

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, potentially leading to vanishing or exploding gradients, along with high memory usage with complexity $O(t)$.

Alternatively in *teacher forcing loss* the past sequence $w_{1:t}$ uses ground truth tokens before step $t$. This allows for parallel computation of the loss across all positions ($O(1)$ complexity) and therefore a single forward/backward pass through the model, which stabilize the gradients.

In the proposed model architecture only trainable parameters are projection layer weights $W_p$ and biases $b_p$, along with transformer decoder parameters $\theta_{dec}$. The projection layer parameters contribute to the loss gradient as

$$\frac{\partial\mathcal{L}}{\partial W_p} = \sum_{t=1}^{T} \left( \frac{\partial\mathcal{L}_t}{\partial z_t} W_e \right) \frac{\partial y_t}{\partial v} \frac{\partial v}{\partial W_p} \quad (15)$$

For each decoder layer $l$, the decoder layer gradient is

$$\frac{\partial\mathcal{L}}{\partial\theta^l_{dec}} = \sum_{t=1}^{T} \frac{\partial\mathcal{L}}{\partial h^l_t} \left( \frac{\partial h^l_t}{\partial\theta^l_{dec}} + \sum_{k>t} \frac{\partial h^l_k}{\partial h^l_t} \frac{\partial h^l_t}{\partial\theta^l_{dec}} \right) \quad (16)$$

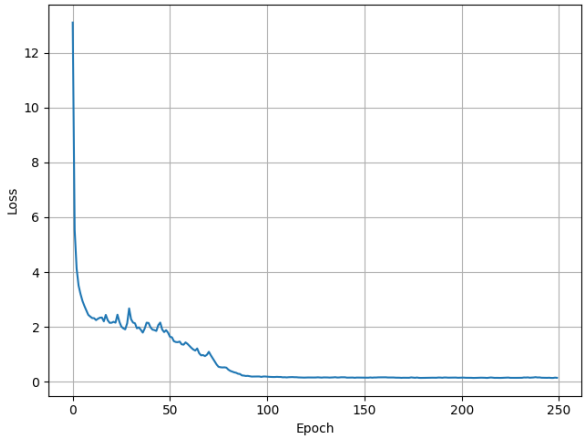



Figure 4: Loss function during initial training of the model

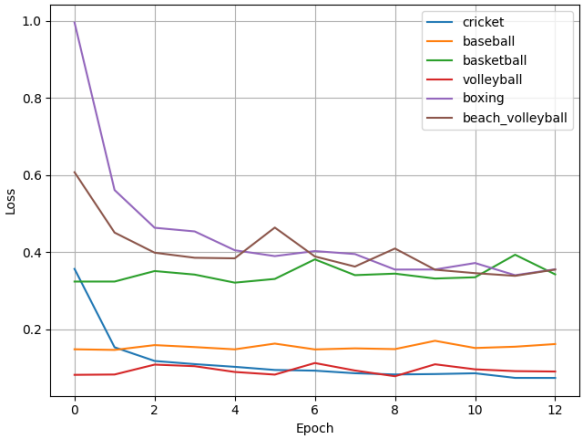



Figure 5: Loss function during fine tuning of the model by category

## 5 Results

The Figure 4 illustrates the evolution of the loss function over 250 training epochs during the initial phase of model training over limited subset. At the outset, the loss is quite high—above 12 is typical for randomly initialized or newly adapted models that have not yet learned meaningful representations from the data. In the first 20 epochs, there is a rapid and substantial decrease in loss, indicating that the model is quickly capturing the most salient patterns and features present in the training set. This steep initial decline is characteristic of effective learning and suggests that the optimizer and learning rate are well-chosen for the early phase.

Beyond the first 20–30 epochs, the loss curve begins to flatten, with smaller oscillations and a gradual downward trend until around epoch 100. After this point, the loss stabilizes at a low value and remains nearly constant for the remainder of training. This plateau suggests that the model has converged and further training does not yield significant improvements in the loss. The absence of upward spikes or instability in the latter part of the curve also indicates that the training process is stable and not suffering from overfitting or catastrophic forgetting. Overall, the figure demonstrates a successful training regime where the model efficiently learns from the data and reaches a steady state of performance.

The Figure 5 presents the loss curves for several sports categories—cricket, baseball, basketball, volleyball, boxing, and beach volleyball—during the fine-tuning phase of a model. Each curve represents the loss value for a specific category as training progresses over 12 epochs. Across all categories, there is a clear trend of rapid loss reduction within the first few epochs, indicating that the model quickly adapts to the new data and learns category-specific features efficiently. Categories like beach volleyball and boxing start with higher initial losses but show significant improvement early on, while others such as cricket and baseball begin with lower losses and stabilize quickly.

After the initial sharp decline, the loss curves for most categories plateau, with only minor fluctuations, suggesting that the model reaches a stable state and further training yields diminishing returns. The variability in final loss values among categories may reflect differences in data complexity, sample size, or intra-class variability. For example, basketball and volleyball maintain higher loss values compared to cricket and baseball, potentially indicating greater difficulty in distinguishing features or less training data for those categories. Overall, the figure demonstrates effective fine-tuning, with the model achieving rapid convergence and stable performance across diverse sports categories.

Table 1 presents the BLEU scores (from 1-gram to 4-gram) for generation performance on two models, MIC21 and Gemma 3. Across all BLEU metrics, Gemma 3 outperforms MIC21, with particularly notable margins at higher n-gram levels: BLEU-1 is 0.79 for Gemma 3 versus 0.71 for MIC21, and BLEU-4 is 0.43 compared to 0.39. This consistent advantage suggests that Gemma 3 produces text that more closely matches the reference outputs, both in terms of individual word choice and longer phrase structure, indicating greater fluency and accuracy in its generated sequences.

| Test | MIC21 | Gemma 3 |
|---|---|---|
| BLEU-1 | 0.71 | 0.79 |
| BLEU-2 | 0.53 | 0.64 |
| BLEU-3 | 0.47 | 0.53 |
| BLEU-4 | 0.39 | 0.43 |

Table 1: Generation Performance Metrics

## 6 Conclusion

The paper presents a scalable framework for domain-adapted multimodal learning, addressing the critical challenge of aligning specialized visual semantics with language generation. By integrating a domain-adapted Detectron2 model with BERT via a trainable projection layer, the architecture enables precise cross-modal adaptation while maintaining computational efficiency. The system leverages the MIC21 corpus—a richly annotated dataset extending COCO categories to over 700 classes across 130 thematic subdomains—to bridge the gap between generic visual recognition and domain-specific linguistic understanding. Key innovations include freezing pre-trained components and focusing training on lightweight projection and transformer decoder layers, achieving stable convergence and efficient resource utilization.

Experimental results demonstrate the framework's effectiveness. Loss curves reveal rapid initial learning followed by stable convergence, indicating robust feature alignment. Fine-tuning performance varies across domains, with sports categories like cricket and baseball achieving lower final losses compared to basketball and volleyball, reflecting differences in intra-class variability and annotation quality. In generation tasks, the model achieves competitive BLEU scores, though it lags behind larger models like Gemma 3, highlighting a trade-off between specialization and generalizability. These results validate the framework's ability to generate context-aware descriptions while preserving domain-specific visual semantics.

A key extension to this study will be a comparison to state of the art models for generation and tagging tasks with an aim to highlight the domain-specificity or trained models.

Future work could extend this approach by expanding the MIC21 ontology to include dynamic interactions between objects and integrating contrastive or adversarial alignment strategies to further refine cross-modal representations. Additionally, exploring efficient attention mechanisms or multilingual extensions of the projection layer could enhance scalability. This work lays a foundation for applications requiring fine-grained multimodal understanding, from medical imaging to industrial inspection, where domain-specific visual-language alignment is critical.

## 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/.

The author is thankful to the reviewers for their insightful and valuable comments.

## 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.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312.

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.