viPolyQwen: Synergizing Prefix-Guided Dynamic Loss Optimization and Attention Pooling for Unified Multimodal Embeddings

Nguyen Anh Nguyen*, **EraX AI Team**, **AI Technology Team, Gtel Mobile JSC (GMobile)** * Corresponding Author: nguyen@hatto.com

(Draft - Work in Progress - Empirical Results Pending)

Abstract

Multimodal representation learning strives to bridge the semantic gap between disparate data types like text and images. While Vision-Language Models (VLMs) have advanced this frontier, generating unified embeddings that are both versatile across diverse tasks (similarity, retrieval, QA) and computationally efficient remains a significant challenge. Existing paradigms often resort to task-specific models, separate embedding spaces, or complex multi-vector architectures, hindering seamless integration and potentially increasing system latency. We introduce ***viPolyQwen****, a novel approach for learning a single, high-dimensional (1024-d), unified multimodal embedding space. Leveraging the expressive power of the Qwen2-VL-2B-Instruct foundation model, viPolyQwen is trained using a unique combination of: (1) an expansive, highly heterogeneous dataset (>11 million samples) encompassing five distinct multimodal interaction types (text similarity, instruction following, OCR, single/multi-turn VQA), with a strong focus on Vietnamese alongside multilingual data; (2) a* ***prefix-guided dynamic mixed-loss optimization strategy*** *that explicitly conditions the learning process, tailoring the contrastive objective function (InfoNCE, Triplet Margin, MSE Regression, Cosine Similarity) on a per-sample basis during training; and (3) an* ***Attention Pooling*** *mechanism that dynamically aggregates information from the VLM encoder's output sequence, prioritizing salient features to generate richer, more context-aware 1D embeddings compared to conventional pooling methods. We posit that this synergistic approach yields a powerful yet architecturally simpler embedding model, potentially streamlining demanding applications like multimodal RAG and cross-modal analysis, particularly for complex, text-rich visual inputs, offering a distinct alternative to single-objective or multi-vector paradigms.*

1. Introduction

The deluge of multimodal information necessitates AI systems capable of holistically understanding and reasoning across text, vision, and structured data. A cornerstone of such systems is the ability to represent diverse inputs within a shared, meaningful vector space $\mathcal{E}$, facilitating tasks like semantic search (k-NN search in $\mathcal{E}$), cross-modal retrieval, recommendation, and Retrieval-Augmented Generation (RAG) [Lewis et al., 2020]. While large Vision-Language Models (VLMs) [Radford et al., 2021; Bai et al., 2023; Alayrac et al., 2022] have demonstrated remarkable capabilities in aligning vision and language, translating their internal representations into effective, general-purpose embeddings $\mathbf{e} \in \mathcal{E}$ presents several challenges.

Firstly, fine-tuning VLMs often yields embeddings specialized for a single task

objective $\mathcal{L}_{\text{task}}$ (e.g., image-text contrastive loss in CLIP [Radford et al., 2021]). While effective for that specific task, these embeddings may be suboptimal for others with different geometric requirements in $\mathcal{E}$ (e.g., fine-grained text similarity regression or visual question answering grounding). This can necessitate maintaining multiple specialized models, increasing operational complexity.

Secondly, representing complex, structured inputs like documents often leads to multi-vector approaches [Faysse et al., 2024; Zhang et al., 2023]. These methods decompose the input into multiple representations (e.g., global context $\mathbf{e}_{global}$, local patches $\{\mathbf{e}_{local,i}\}$). While potentially capturing finer granularity, they introduce significant downstream complexity, requiring specialized indexing structures and multi-stage retrieval algorithms (e.g., ColBERT-style late interaction [Khattab & Zaharia, 2020]) that deviate from standard, highly optimized dense vector search paradigms (like FAISS [Johnson et al., 2019]).

Thirdly, the mechanism used to pool the sequence of VLM encoder outputs $\mathbf{H} \in \mathbb{R}^{N \times D_{hidden}}$ into a single vector $\mathbf{c} \in \mathbb{R}^{D_{hidden}}$ profoundly impacts the final embedding quality. Standard strategies like mean pooling ($\mathbf{c}_{\text{mean}} = \frac{1}{N}\sum \mathbf{h}_i$) risk diluting salient information, while last-token pooling ($\mathbf{c}_{\text{last}} = \mathbf{h}_N$) ignores potentially crucial context from earlier in the sequence. This is particularly detrimental for information-dense inputs like documents or images containing embedded text, where critical features might be localized and averaged out or simply missed.

To address these shortcomings, we propose **viPolyQwen**, a unified multimodal embedding model built upon Qwen2-VL-2B-Instruct [Bai et al., 2023]. Our approach aims to generate a single 1024-dimensional vector $\mathbf{e} \in \mathbb{R}^{1024}$ capable of representing diverse multimodal inputs effectively. Its design is driven by three core principles:

1. **Highly Diverse Multi-Task Training Data:** We curate and utilize a large-scale dataset ($D = \{(x_i, y_i, \text{type}_i, ...)\}_{i=1}^{M}$, $M > 11 \times 10^6$) incorporating five distinct data formats (`type`) and associated tasks: text similarity pairs (with scores s_i), instruction-following sequences, Optical Character Recognition (OCR) / Optical Character Questioning (OCQ), single-turn Visual Question Answering (VQA), and multi-turn VQA. This diversity, with a focus on Vietnamese and substantial multilingual components, fosters robustness and generalization.
2. **Prefix-Guided Dynamic Loss Optimization:** We introduce an explicit conditioning mechanism during training. Task-specific prefixes $p_i \in \{\texttt{<ocr>}, \texttt{<text\_pair>}, ...\}$ are prepended to the input x_i. This prefix p_i serves as a discrete signal that dynamically selects a tailored objective function $\mathcal{L}_{\text{type}(p_i)}$ (composed of InfoNCE, Triplet Margin, MSE, Cosine Similarity components) specifically optimized for that task structure. This allows the model, represented by parameters θ, to learn task-aware representations within the unified space $\mathcal{E}$.
3. **Attention Pooling for Richer Embeddings:** Departing from standard pooling, we employ a learnable Attention Pooling mechanism (Section

3.2) over the final hidden state sequence $\mathbf{H}$. This allows the model to dynamically identify and weight the most salient textual and visual features (α_i weights for $\mathbf{h}_i$), producing a more informative and contextually relevant intermediate representation $\mathbf{c} = \sum \alpha_i \mathbf{h}_i$, crucial for capturing nuances in complex inputs before projection to the final embedding $\mathbf{e}$.

We hypothesize that the synergy between diverse multi-task learning, explicit prefix-guided dynamic loss adaptation, and attention-based feature aggregation enables viPolyQwen to produce unified 1D embeddings that are both powerful for downstream tasks and significantly simpler architecturally and computationally to deploy compared to multi-vector or purely task-specific paradigms. This work was undertaken in collaboration with the AI technology team at Gtel Mobile JSC (GMobile), whose support was instrumental.

2. Related Work

Our work builds upon and distinguishes itself from several lines of research:

- **Multimodal Contrastive Learning (e.g., CLIP, ALIGN):** Foundational models like CLIP [Radford et al., 2021] and ALIGN [Jia et al., 2021] excel at learning image-text alignment through a single, powerful contrastive objective $\mathcal{L}_{\text{contrastive}}$ across vast web-scale datasets. However, this single objective, while effective for retrieval, may not optimally capture the nuances required for diverse downstream tasks like fine-grained semantic similarity regression (requiring MSE-like loss) or structured QA grounding (benefiting from margin-based losses like Triplet) within the *same* embedding space. Adapting these models often requires further task-specific fine-tuning, potentially leading to multiple specialized models or compromising the original general alignment. viPolyQwen explicitly addresses this by incorporating multiple loss formulations within a single training framework, guided by task type.
- **Sentence & Text Embeddings (e.g., Sentence-BERT):** Fine-tuning approaches like Sentence-BERT [Reimers & Gurevych, 2019] typically focus on optimizing for a specific pair-based task structure (e.g., semantic similarity using NLI data or regression on STS benchmarks). Applying such a focused approach naively to multimodal, multi-task data risks creating embeddings biased towards one structure, potentially degrading performance on others (e.g., an embedding optimized solely for image-caption similarity might not be ideal for VQA reasoning). viPolyQwen's dynamic loss selection avoids this bias by applying the appropriate optimization pressure for each data type encountered.
- **Document AI & Multi-Vector Representations (e.g., ColPali):** Addressing the complexity of structured documents, multi-vector approaches like ColPali [Faysse et al., 2024] dedicate separate representations for different granularities (e.g., global context + local patches via Pali-3). While potentially capturing fine-grained detail, this necessitates specialized retrieval mechanisms like ColBERT-style late interaction [Khattab & Zaharia, 2020], which involve token-level similarity computations and

aggregation, deviating significantly from standard, highly efficient vector search (e.g., using ANN libraries like FAISS [Johnson et al., 2019]). Our prefix-guided approach, coupled with Attention Pooling, offers an alternative hypothesis: a *single* vector can be imbued with sufficient task-awareness and salient feature representation to handle diverse tasks effectively, thereby retaining architectural simplicity. The prefix explicitly conditions the *learning* process, aiming to encode task-relevant nuances directly into the unified embedding, while Attention Pooling helps capture local salience without resorting to separate vectors.

- **Pooling Mechanisms:** While mean/max/last-token pooling are computationally cheap, they are often suboptimal information aggregators. Self-attention pooling [Lin et al., 2017] adds complexity. Our simpler learnable context vector approach for Attention Pooling (Section 3.2) provides a balance, enabling dynamic weighting without full self-attention overhead.
- **Multi-Task Learning & Dynamic Loss:** Training models on multiple tasks simultaneously can improve generalization [Caruana, 1997]. Dynamically selecting or weighting losses is known to help navigate conflicting gradient signals [Kendall et al., 2018; Chen et al., 2018]. Our prefix-guided mechanism provides an *explicit, discrete* signal for selecting pre-defined, task-optimized loss combinations, differing from methods that learn continuous loss weights or rely on implicit task inference. This explicit signal ensures the correct geometric constraints are applied during optimization for each sample type.
- **Vietnamese & Cross-Lingual Models:** We specifically address the need for high-quality multimodal embeddings for Vietnamese, leveraging substantial native data alongside multilingual resources to foster both strong in-language performance and zero-shot cross-lingual capabilities [Conneau et al., 2019].

In summary, viPolyQwen's unique contribution lies in the deliberate synergy of: (1) harnessing a powerful VLM backbone, (2) explicitly conditioning the learning process on diverse task structures via prefix signals coupled with dynamic loss selection, and (3) employing Attention Pooling to generate a rich, unified 1D embedding. This combination aims to circumvent the limitations of single-objective training (like CLIP), the task bias of simple fine-tuning (like Sentence-BERT style), and the architectural complexities of multi-vector representations (like ColPali).

3. Methodology

(Sections 3.1 Model Architecture, 3.2 Attention Pooling Mechanism, 3.3 Projection and Normalization remain largely the same as the previous version, with detailed LaTeX formulas already included. Ensure consistency.)

3.4 Prefix-Guided Input Representation & Conditioning (Training)

During training, the `MixedBatchCollator` (`mix_data_collator (5).py`) pre-

processes each sample $(x_i, y_i, \text{type}_i, ...)$ from the dataset D. Crucially, based on the specified `data_type`, a corresponding discrete prefix token $p_i \in P = \{$`<ocr>`, `<text_pair>`, `<instr>`, `<vqa_single>`, `<vqa_multi>`$\}$ is prepended to the primary textual component of the input x_i. This modified input, denoted $x'_i = (\text{prefix}(p_i), x_i)$, is then fed into the model.

This explicit prefix p_i serves as a **conditioning signal**. Let the embedding function parameterized by θ be $f_\theta : (X', P) \mapsto \mathcal{E}$, where X' is the space of possibly prefixed inputs. The prefix p_i directly influences the selection of the loss function $\mathcal{L}_{\text{type}(p_i)}$ used for the i-th sample (Section 4.2). Consequently, the gradient contributing to the update of the shared parameters θ is task-dependent:

$$\nabla_\theta \mathcal{L}_{\text{batch}} = \frac{1}{B} \sum_{i=1}^{B} \nabla_\theta \mathcal{L}_{\text{type}(p_i)}(f_\theta(x'_i), f_\theta(y'_i))$$

This explicit conditioning allows the shared parameters θ to learn representations that are sensitive to the demands of different tasks, effectively enabling task specialization *within* the unified embedding space $\mathcal{E}$, rather than requiring separate models or fragmented spaces. For inference on general data where the task is unknown or simply content representation is desired, no prefix is used ($p = \text{None}$), and the model produces a general-purpose embedding based on its aggregated learned knowledge.

4. Training Paradigm

(Section 4.1 Dataset Composition remains the same.)

4.2 Prefix-Guided Dynamic Mixed-Loss Optimization

The cornerstone of our training objective is the dynamic application of task-specific loss functions, orchestrated by the `multi_purpose_contrastive_loss` function (`losses (6).py`) based on the explicit prefix signal p_i. Let $(\mathbf{e}_{a,i}, \mathbf{e}_{b,i})$ be the L2-normalized embeddings derived from the i-th input pair (x'_i, y'_i) via the model f_θ, including the Attention Pooling step. The specific loss applied, $\mathcal{L}_{\text{type}(p_i)}$, is chosen from a suite of functions designed to impose appropriate geometric constraints in the embedding space $\mathcal{E}$ for each task:

- **For $p_i =$ `<text_pair>` (Similarity Regression & Contrastive):** Aims to align embedding distance with semantic similarity score $s_i \in [0, 1]$ while pushing dissimilar pairs apart.

  $$\mathcal{L}_{\text{text\_pair}} = \lambda_{nce} \mathcal{L}_{NCE}(\mathbf{e}_{a,i}, \mathbf{e}_{b,i}, T) + \lambda_{mse} \mathcal{L}_{MSE}(\mathbf{e}_{a,i}, \mathbf{e}_{b,i}, s_i)$$

  where T is the InfoNCE temperature (0.07), $\mathcal{L}_{NCE}$ is the symmetric InfoNCE loss promoting discrimination within the batch, and $\mathcal{L}_{MSE} = (\frac{1}{2}(\mathbf{e}_{a,i}^T \mathbf{e}_{b,i} + 1) - s_i)^2$ encourages the cosine similarity (mapped to [0,1]) to match the ground truth score.

- **For $p_i =$ `<instr>` (Instruction Grounding):** Aims to maximize alignment between instruction and output embeddings.

$$\mathcal{L}_{\text{instr}} = \lambda_{nce}\mathcal{L}_{NCE}(\mathbf{e}_{a,i}, \mathbf{e}_{b,i}, T) + \lambda_{cos}\mathcal{L}_{Cos}(\mathbf{e}_{a,i}, \mathbf{e}_{b,i})$$

  where $\mathcal{L}_{Cos} = (1 - \mathbf{e}_{a,i}^T \mathbf{e}_{b,i})$ directly minimizes the angle between the pair.

- **For $p_i \in \{$`<ocr>`, `<vqa_single>`, `<vqa_multi>`$\}$ (Question Answering Grounding):** Employs a margin-based loss to ensure the embedding of the input (image+question) is closer to its correct answer embedding than to embeddings of answers for other inputs (hard negatives) by at least a margin m.

$$\mathcal{L}_{\text{ocr/vqa}} = \lambda_{nce}\mathcal{L}_{NCE}(\mathbf{e}_{a,i}, \mathbf{e}_{b,i}, T) + \lambda_{trip}\mathcal{L}_{Triplet}(\mathbf{e}_{a,i}, \mathbf{e}_{b,i}, \mathcal{N}_i, m', T)$$

  where $\mathcal{L}_{Triplet} = \max\left(0, \max_{\mathbf{e}_n \in \mathcal{N}_i} \frac{\mathbf{e}_{a,i}^T \mathbf{e}_n}{T} - \frac{\mathbf{e}_{a,i}^T \mathbf{e}_{b,i}}{T} + m'\right)$, $\mathcal{N}_i = \{\mathbf{e}_{b,j} \mid j \neq i\}$ is the set of in-batch negatives, and m' is the effective margin (potentially adjusted for multi-turn VQA, e.g., $m' = 1.5m$ with $m = 0.2$).

The hyperparameters $\lambda_{(\cdot)}$ control the relative contribution of each loss component (implicitly set to 1.0 currently). The overall batch loss is $\mathcal{L}_{\text{batch}} = \frac{1}{B}\sum_{i=1}^{B} \mathcal{L}_{\text{type}(p_i)}$. This dynamic loss selection, guided explicitly by the prefix, is crucial for navigating the potentially competing geometric objectives of different tasks within the unified embedding space $\mathcal{E}$, enabling the model to learn specialized representations without parameter fragmentation.

(Section 4.3 Implementation Details remains the same.)

5. Expected Performance and Discussion

(Keep the previous content but enhance the contrasts):

- **Task-Aware Unified Embeddings:** Unlike models trained with a single objective (e.g., CLIP's contrastive loss) which might struggle with tasks requiring different geometric properties (like similarity regression), or models needing separate vectors for different aspects (e.g., ColPali), viPolyQwen aims for task versatility *within* its single vector. The explicit prefix-guided training allows it to learn representations sensitive to diverse task demands (e.g., fine-grained semantics for `<text_pair>`, spatial/object/text grounding for `<vqa>`), creating a more flexible unified embedding compared to single-objective models, without the architectural overhead of multi-vector approaches.
- **Enhanced Retrieval for Text-in-Image:** ... (Keep explanation of Attention Pooling benefit) ... This contrasts with mean pooling's potential dilution and multi-vector methods' need to explicitly retrieve and manage patch embeddings.
- **Simplified System Architecture:** ... (Keep comparison to ColPali, emphasizing standard vector DB compatibility) ... This advantage stems

directly from producing a single, albeit task-informed, vector representation.

- **Improved Cross-Modal Reasoning:** ... (Keep existing point) ...
- **Strong Vietnamese Performance:** ... (Keep existing point) ...
- **Versatility & Inference Strategy:** ... (Expand slightly) ... The unified embedding **e** serves diverse downstream needs. For general content representation and retrieval (text chunks, images, image+description), embeddings are generated *without* prefixes. For specific task-oriented queries *where the goal aligns directly with a trained task structure* (e.g., querying based on an OCR question about an image), using the corresponding prefix (`<ocr>`) *may* yield more relevant results *if the indexed database also contains embeddings generated under similar task conditioning.* However, the more common and robust approach for complex tasks like document VQA involves a two-stage process: (1) Use prefix-less embeddings for efficient retrieval of relevant context (documents/images) from a standard vector DB. (2) Feed the retrieved context and the original query (potentially with prefix) into the VLM's generative component (or another generative model) to synthesize the final answer. Our model excels at the crucial first stage by providing high-quality, unified context embeddings.

Table 1: Text-Image Retrieval Benchmarks (Zero-Shot Evaluation Expected)

Table 2: Document VQA / Retrieval Benchmarks (Illustrative)

Discussion: The primary trade-off lies between the representational capacity of a single 1024-d vector versus the explicit detail captured by multiple, potentially higher-dimensional vectors. We argue that the sophisticated training and the Attention Pooling mechanism significantly enhance the information density of our single vector, potentially matching or exceeding the practical performance of multi-vector systems in many retrieval scenarios while offering substantial implementation advantages. Empirical validation against methods like ColPali on document retrieval benchmarks (e.g., DUDE [Faysse et al., 2024]) will be critical to substantiate this claim.

6. Conclusion and Future Work viPolyQwen introduces a novel framework for learning unified multimodal embeddings, characterized by its prefix-guided dynamic loss strategy and the use of Attention Pooling. By training a powerful VLM foundation on a highly diverse multi-task dataset, we aim to produce a single 1024-d vector that effectively represents text, images, and their combinations, simplifying downstream applications like RAG and cross-modal search, especially for Vietnamese language and complex visual documents. Attention Pooling is hypothesized to be key in capturing salient details within this single vector, offering a potentially more efficient alternative to multi-vector architectures.

Our immediate future work involves rigorous empirical evaluation:

1. Benchmarking on standard cross-modal retrieval datasets (MS-COCO, Flickr30k).

2. Evaluating performance on Vietnamese-specific tasks (e.g., ViTextEval, UIT-ViQuAD for retrieval context).
3. Assessing document retrieval capabilities (e.g., DUDE) and comparing against relevant baselines, including multi-vector methods.
4. Conducting ablation studies to isolate the contributions of Attention Pooling versus mean/last-token pooling and the impact of different loss components/data types.
5. Releasing the model checkpoints, evaluation scripts, and usage guidelines.

Further explorations may include scaling to larger base models and incorporating additional modalities or task types.

7. References

[Alayrac et al., 2022] Jean-Baptiste Alayrac, Jeff Donahue, et al. *Flamingo: a Visual Language Model for Few-Shot Learning.* NeurIPS 2022. [Ba et al., 2016] Jimmy Lei Ba, Jamie Ryan Kiros, Geoffrey E. Hinton. *Layer Normalization.* arXiv:1607.06450, 2016. [Bahdanau et al., 2014] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate.* ICLR 2015 (arXiv:1409.0473). [Bai et al., 2023] Jinze Bai, Shuai Bai, et al. *Qwen-VL: A Versatile Vision-Language Model for Understanding, Localization, Text Reading, and Beyond.* arXiv:2308.12966, 2023. [Caruana, 1997] Rich Caruana. *Multitask Learning.* Machine Learning, 28(1):41–75, 1997. [Chen et al., 2018] Yu-Tian Chen, Yan-Kai Wang, et al. *GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks.* ICML 2018. [Chen et al., 2022] Xi Chen, Xiao Wang, et al. *PaLI: A Jointly-Scaled Multilingual Vision-Language Model.* arXiv:2209.06794, 2022. [Conneau et al., 2019] Alexis Conneau, Kartikay Khandelwal, et al. *Unsupervised Cross-lingual Representation Learning at Scale.* ACL 2020 (arXiv:1911.02116). [Devlin et al., 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* NAACL 2019 (arXiv:1810.04805). [Faysse et al., 2024] Manuel Faysse, Hugues Sibille, et al. *ColPali: Efficient Document Retrieval with Vision Language Models.* arXiv:2407.01449, 2024. [Huang et al., 2022] Yupan Huang, Tengchao Lv, et al. *LayoutLMv3: Pre-training for Document AI with Unified Text and Image Masking.* ACM Multimedia 2022 (arXiv:2204.08387). [Jia et al., 2021] Chao Jia, Yinfei Yang, et al. *Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision.* ICML 2021. [Johnson et al., 2019] Jeff Johnson, Matthijs Douze, Hervé Jégou. *Billion-scale similarity search with GPUs.* IEEE Transactions on Big Data, 2019 (arXiv:1702.08734). [Kendall et al., 2018] Alex Kendall, Yarin Gal, Roberto Cipolla. *Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics.* CVPR 2018. [Khattab & Zaharia, 2020] Omar Khattab and Matei Zaharia. *ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT.* SIGIR 2020. [Lewis et al., 2020] Patrick Lewis, Ethan Perez, et al. *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.* NeurIPS 2020. [Li et al.,

2022] Junnan Li, Dongxu Li, Caiming Xiong, Steven Hoi. *BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation.* ICML 2022. [Lin et al., 2017] Zhouhan Lin, Minwei Feng, et al. *A Structured Self-attentive Sentence Embedding.* ICLR 2017. [Loshchilov & Hutter, 2017] Ilya Loshchilov, Frank Hutter. *Decoupled Weight Decay Regularization.* ICLR 2019 (arXiv:1711.05101). [Radford et al., 2021] Alec Radford, Jong Wook Kim, et al. *Learning Transferable Visual Models From Natural Language Supervision.* ICML 2021. [Reimers & Gurevych, 2019] Nils Reimers and Iryna Gurevych. *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.* EMNLP 2019. [Xu et al., 2020] Yiheng Xu, Minghao Li, et al. *LayoutLM: Pre-training of Text and Layout for Document Image Understanding.* KDD 2020. [Yu et al., 2022] Jiahui Yu, Zirui Wang, et al. *CoCa: Contrastive Captioners are Image-Text Foundation Models.* arXiv:2205.01917, 2022. [Yuan et al., 2021] Lu Yuan, Dongdong Chen, et al. *Florence: A New Foundation Model for Computer Vision.* arXiv:2111.11432, 2021. [Zhang et al., 2023] Zheng Zhang, Jonas Müller, et al. *Beyond Pixels and Patches: Utilizing VLM for Document Information Extraction.* arXiv:2310.00425, 2023. [ViPolyQwen Repo, 2024] Steve Nguyen Anh Nguyen, et al. *viPolyQwen GitHub Repository.* https://github.com/EraX-AI/viPolyQwen, 2024.