UNIVERSIDADE FEDERAL DE GOIÁS / INSTITUTO DE INFORMÁTICA

Modelos Multimodais para Recuperação Semântica

Fine-Tuning Comparativo de Modelos em Múltiplos Domínios

Julia Soares Dollis

UNIVERSIDADE FEDERAL DE GOIÁS (UFG)
INSTITUTO DE INFORMÁTICA (INF)

JULIA SOARES DOLLIS

Modelos Multimodais para Recuperação Semântica

Fine-Tuning Comparativo de Modelos em Múltiplos Domínios

Goiânia
2025

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TRABALHO DE CONCLUSÃO DE CURSO DE GRADUAÇÃO NO REPOSITÓRIO INSTITUCIONAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio do Repositório Institucional (RI/UFG), regulamentado pela Resolução CEPEC no 1240/2014, sem ressarcimento dos direitos autorais, de acordo com a Lei no 9.610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo dos Trabalhos de Conclusão dos Cursos de Graduação disponibilizado no RI/UFG é de responsabilidade exclusiva dos autores. Ao encaminhar(em) o produto final, o(s) autor(a)(es)(as) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do Trabalho de Conclusão de Curso de Graduação (TCCG)

Nome(s) completo(s) do(a)(s) autor(a)(es)(as): JULIA SOARES DOLLIS

Título do trabalho: Modelos Multimodais para Recuperação Semântica

Fine-Tuning Comparativo de Modelos em Múltiplos Domínios

2. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador) Concorda com a liberação total do documento [X] SIM [ ] NÃO[1]

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante: a) consulta ao(à)(s) autor(a)(es)(as) e ao(à) orientador(a); b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo do TCCG. O documento não será disponibilizado durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;

- Submissão de artigo em revista científica;

- Publicação como capítulo de livro.

Obs.: Este termo deve ser assinado no SEI pelo orientador e pelo autor.

Documento assinado eletronicamente por **Julia Soares Dollis**, **Discente**, em 16/03/2026, às 12:11, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020.

Documento assinado eletronicamente por **Fernando Marques Federson**, **Professor do Magistério Superior**, em 21/03/2026, às 09:35, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do Decreto nº 10.543, de 13 de novembro de 2020 .

JULIA SOARES DOLLIS

Modelos Multimodais para Recuperação Semântica

Fine-Tuning Comparativo de Modelos em Múltiplos Domínios

Relatório final de Trabalho de Conclusão de Curso, apresentado à Universidade Federal de Goiás, como parte das exigências para a obtenção do título de Bacharel em Inteligência Artificial.

Orientador: Prof. Dr. Fernando Marques Federson

Goiânia

2025

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UFG.

DOLLIS, JULIA SOARES
Modelos Multimodais para Recuperação Semântica [manuscrito]: Fine-Tuning Comparativo de Modelos em Múltiplos Domínios / JULIA SOARES DOLLIS. - 2025.
170 f.: 2025

Orientador: Prof. Dr. Fernando Marques Federson
Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Goiás, Instituto de Informática (INF), Inteligência Artificial, Goiânia, 2025.

1. Inteligência Artificial. 2. Modelos Texto–imagem. 3. Recuperação Semântica.

I. Federson, Fernando Marques , orient. II. Título.

CDU 004

JULIA SOARES DOLLIS

Modelos Multimodais para Recuperação Semântica
Fine-Tuning Comparativo de Modelos em Múltiplos Domínios

Relatório final de Trabalho de Conclusão de Curso, apresentado à Universidade Federal de Goiás, como parte das exigências para a obtenção do título de Bacharel em Inteligência Artificial.

Data da Aprovação: 09 de dezembro de 2025.

Prof. Dr. Fernando Marques Federson
Orientador (INF-UFG)

Prof. Dr. Aldo André Díaz Salazar
Coordenador de TCC do BIA (INF-UFG)

Prof. Dr. Anderson da Silva Soares
Coordenador do BIA (INF-UFG)

Prof. Dr. Sávio Salvarino Teles de Oliveira
(INF-UFG)

JULIA SOARES DOLLIS

Modelos Multimodais para Recuperação Semântica

Fine-Tuning Comparativo de Modelos em Múltiplos Domínios

RESUMO

Este Relatório de Conclusão de Curso tem como objetivo reunir os resultados da minha jornada para me tornar um especialista em **Modelos Multimodais**. Uma ilustração e sua narrativa descrevem os períodos de trabalho. Os Apêndices contêm os Termos de Aceite de Entrega e os resultados obtidos durante cada período de trabalho.

Palavras-chave: Inteligência artificial; Modelos texto–imagem; Recuperação semântica.

ABSTRACT

This Course Completion Report aims to bring together the results of my journey to become an expert in **Multimodal Models**. An illustration and its narrative describe the work periods. The Appendices contain the Delivery Acceptance Terms and the results obtained during each work period.

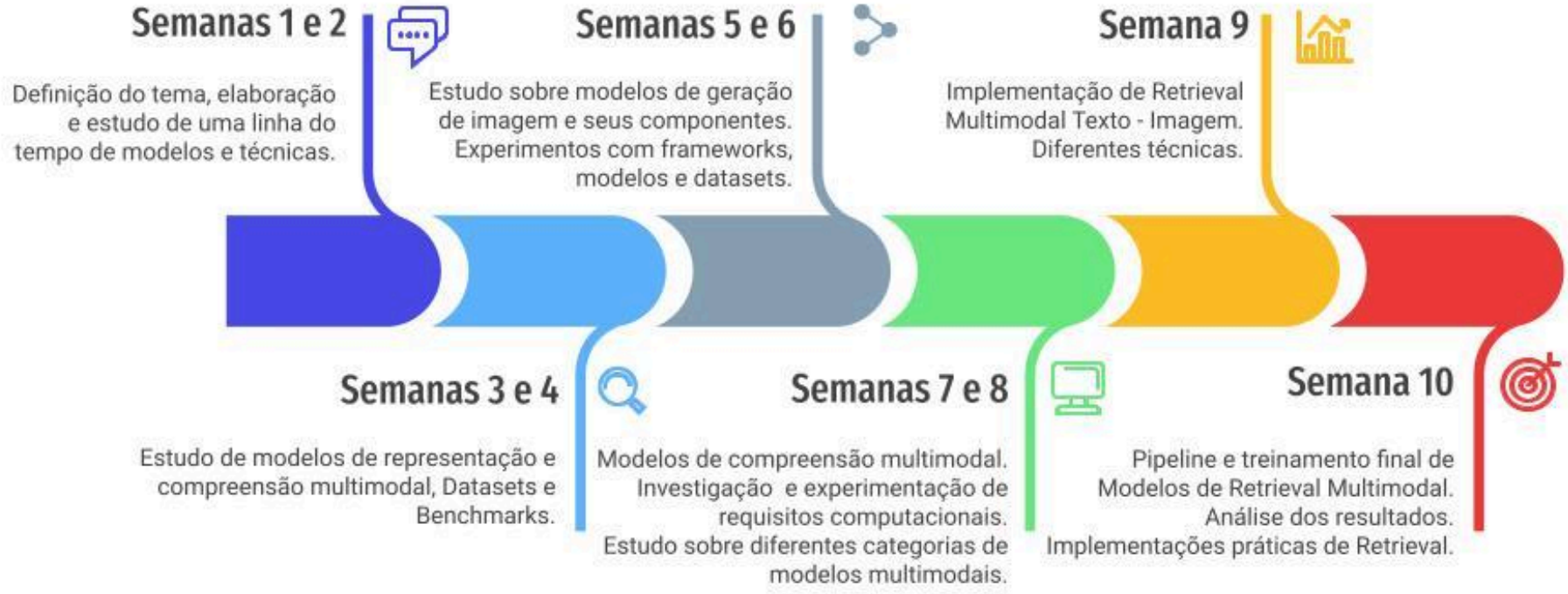
Keywords: Artificial intelligence; Text-image templates; Semantic retrieval.

Goiânia

2025

Minha Jornada

Julia Soares Dollis
Especialista em: Modelos Multimodais



MINHA JORNADA

Nome: **Julia Soares Dollis**

Especialidade: **Modelos Multimodais**

Objetivo deste documento

Durante o processo da disciplina Residência em IA[1], foram gerados diversos resultados na construção da minha especialização. A cada semana, um conjunto de resultados foi formalizado por um Termo de Aceite de Entrega e avaliado por uma banca, considerando o planejado e o realizado para o período. Este documento tem como objetivo descrever esses resultados obtidos, fazendo referência aos Termos de Aceite de Entrega e seus documentos associados.

Minha Jornada

Minha Jornada começou nas **Semanas 1 e 2** com a definição e delimitação do tema central da minha especialização: **Modelos Multimodais**, com foco específico na relação texto–imagem. Diante da amplitude do tema, iniciei meus estudos consultando artigos do tipo *survey* como "*Unified multimodal understanding and generation models: Advances, challenges, and opportunities*"[2] e repositórios de surveys no GitHub, o que me permitiu categorizar as principais subdivisões da área e organizar uma visão estruturada do "estado da arte". A partir dessas referências iniciais, elaborei uma categorização própria, agrupando famílias de modelos e suas abordagens metodológicas. Em seguida, a partir da leitura direta dos artigos, consolidei esse conhecimento por meio de tabelas comparativas que analisam arquiteturas, dados de treino, funções de perda e outras características essenciais, além de mapear benchmarks amplamente utilizados. Também realizei uma curadoria sistemática de

[1] Dez Semanas, entre setembro de 2025 e dezembro de 2025.

[2] Zhang, Xinjie, et al. "*Unified multimodal understanding and generation models: Advances, challenges, and opportunities*." *arXiv preprint arXiv:2505.02567* (2025).

trabalhos fundamentais, desde sobre os modelos basilares, como o CLIP[3], até os artigos mais recentes, construindo uma lista cronológica de leitura para compreender tanto a evolução histórica quanto as bases conceituais da área. Todo esse material, incluindo listas de artigos, ferramentas estudadas, comparativos e fichamentos, foi organizado no **Apêndice 1**. Ao final dessas duas **Semanas**, eu havia estabelecido não apenas o tema da especialização, mas também a base teórica, a metodologia de estudo e a estrutura que orientariam toda a continuidade do meu trabalho.

Nas **Semanas 3 e 4**, avancei do mapeamento geral da área para um foco mais direcionado nos modelos multimodais de natureza autorregressiva, aprofundando tanto a teoria quanto a evolução histórica dessas arquiteturas. Dediquei-me à leitura de trabalhos fundamentais que permitiram construir uma linha do tempo consistente, iniciando por modelos pioneiros, como o DeViSE (2013)[4], e avançando para os mais recentes, como Kosmos-1 (2023)[5]. Paralelamente, conduzi um estudo sobre os principais *datasets* utilizados em pré-treino e *fine-tuning*, analisando a composição e o papel de bases de dados extensas no avanço da área. Outro ponto importante desse período, foi a leitura do *survey* sobre *Text-rich Image Understanding* (TIU)[6], que ampliou minha compreensão sobre como componentes de percepção visual como OCR, detecção de layout e reconhecimento de fórmulas se articulam com mecanismos de raciocínio semântico em *Multimodal Large Language Models* (MLLMs). Essa imersão reforçou a importância de arquiteturas basilares, como a do CLIP, que continuam servindo de base e sendo adaptadas nos modelos atuais. A relação completa dos artigos estudados, acompanhada de resumos técnicos e da análise dos *datasets* consultados, está reunida no **Apêndice 2**. Assim, as **Semanas 3 e 4** representaram um momento essencial de aprofundamento técnico e histórico, no qual compreendi como os modelos multimodais autorregressivos evoluíram, quais conjuntos de

[3] Radford, Alec, et al. "Learning transferable visual models from natural language supervision." *International conference on machine learning*. PmLR, 2021.

[4] Frome, Andrea, et al. "Devise: A deep visual-semantic embedding model." *Advances in neural information processing systems* 26 (2013).

[5] Huang, S., et al. "Language is not all you need: aligning perception with language models. arXiv." *Preprint posted online February* 27 (2023).

[6] Fu, Pei, et al. "Multimodal Large Language Models for Text-rich Image Understanding: A Comprehensive Review." *Findings of the Association for Computational Linguistics: ACL 2025* (2025): 19941-19958.

dados impulsionaram esse avanço e de que forma diferentes arquiteturas contribuíram para a construção das bases que sustentam os modelos contemporâneos.

As **Semanas 5 e 6** foram dedicadas ao aprofundamento nos modelos de geração de imagens e em seus componentes fundamentais, além de uma fase importante de experimentação. Estudei os *surveys* "*Unified Multimodal Understanding and Generation Models: Advances, Challenges, and Opportunities*"[7] e "*MME-Survey: A Comprehensive Survey on Evaluation of Multimodal LLMs*"[8], que ampliaram minha visão sobre as subdivisões entre modelos voltados para compreensão e para geração multimodal, bem como sobre os diferentes paradigmas de geração autorregressiva, como *Pixel-based Encoding*, *Semantic Encoding*, *Learnable Query Encoding* e *Hybrid Encoding*. A leitura desse material evidenciou um ponto essencial no meu aprendizado: compreender o papel dos tokenizadores e dos módulos de quantização que estruturam o espaço visual em muitos modelos atuais. Por isso, dediquei parte dessas semanas ao estudo de VQ-VAE[9] e VQGAN. Na vertente experimental, conduzi testes via API com modelos como Qwen (VL e Omni)[10] para observar diferenças de comportamento e capacidades. Em paralelo, realizei um *fine-tuning* do modelo BLIP[11] utilizando um *dataset* em Português-BR. Mesmo com apenas uma época e cerca de 20 mil exemplos, já pude observar melhorias significativas nas legendas geradas pelo modelo. Também comecei a investigar frameworks específicos para modelos multimodais e seus usos. Todos esses estudos, experimentos, resumos e comparativos encontram-se organizados no **Apêndice 3**.

Nas **Semanas 7 e 8**, aprofundei meus estudos teóricos para obter uma clareza mais sistêmica da área, expandindo o escopo para incluir também os Modelos de Difusão e suas técnicas de *denoising*. Dediquei-me a textos, vídeos e artigos especializados que detalham o processo de adicionar e remover ruído, bem como as diferenças conceituais entre tratar a

[7] Zhang, Xinjie, et al. "Unified multimodal understanding and generation models: Advances, challenges, and opportunities." *arXiv preprint arXiv:2505.02567* (2025).
[8] Fu, Chaoyou, et al. "Mme-survey: A comprehensive survey on evaluation of multimodal llms." *arXiv preprint arXiv:2411.15296*(2024).
[9] Van Den Oord, Aaron, and Oriol Vinyals. "Neural discrete representation learning." *Advances in neural information processing systems* 30 (2017).
[10] Yang, An, et al. "Qwen3 technical report." *arXiv preprint arXiv:2505.09388* (2025).
[11] Li, Junnan, et al. "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation." *International conference on machine learning*. PMLR, 2022.

imagem como uma sequência de *tokens* (característica de abordagens autorregressivas) e tratá-la como uma entidade global reconstruída progressivamente a partir do ruído. Paralelamente, organizei um mapa dos modelos texto–imagem, categorizando-os em compreensão, geração e arquiteturas híbridas, e registrando anotações sobre *pipelines*, componentes recorrentes e técnicas típicas de cada etapa, o que me permitiu consolidar uma visão mais integrada e comparativa da área. Com essa base teórica, iniciei a análise de possíveis experimentos para reprodução prática, avaliando artigos, repositórios e, **principalmente, os requisitos computacionais envolvidos**. Esse enfrentamento técnico, comparando a VRAM necessária, o custo esperado em plataformas na nuvem, o tamanho dos *datasets* e a complexidade dos *pipelines*, foi fundamental para compreender quais modelos seriam viáveis dentro das minhas condições atuais de hardware. Em paralelo, entrei em uma fase de experimentação com objetivo de explorar e procurar entender como diferentes arquiteturas multimodais se comportam. Comecei examinando o LLaVA[12], um dos modelos mais influentes na literatura recente, analisando seu repositório oficial, estrutura de pipeline e componentes internos. Essa investigação foi essencial para compreender como encoders visuais e LLMs são acoplados e para identificar potenciais pontos de adaptação. Em seguida, explorei o LLaVA-Mini[13], uma variante otimizada que me permitiu estudar soluções modernas de eficiência computacional e design de modelos mais leves, sem abrir mão de capacidades multimodais relevantes. Por fim, avancei para o TinyGPT-V. Cada uma dessas explorações - LLaVA, LLaVA-Mini e TinyGPT-V - contribuiu para ampliar minha compreensão prática sobre modelos multimodais e fundamentar com mais segurança a definição do experimento a ser desenvolvido em profundidade. O mapa mental elaborado, juntamente com as análises comparativas dos modelos, a avaliação dos requisitos computacionais e o início da implementação da arquitetura escolhida, estão organizados no **Apêndice 4**.

A **Semana 9** marcou a redefinição estratégica do projeto e o início concentrado da implementação em *Retrieval Multimodal*. Embora tenha realizado testes iniciais com o TinyGPT-V, a análise aprofundada do campo indicou que arquiteturas voltadas à

[12] Liu, Haotian, et al. "Visual instruction tuning." *Advances in neural information processing systems* 36 (2023): 34892-34916.

[13] Zhang, Shaolei, et al. "Llava-mini: Efficient image and video large multimodal models with one vision token." *arXiv preprint arXiv:2501.03895* (2025).

recuperação de informação ofereciam um caminho mais alinhado aos objetivos. Assim, dediquei parte significativa da **Semana** ao estudo sistemático dessa linha de pesquisa, investigando requisitos computacionais, características de *datasets* e métricas centrais como Recall@K, MRR e nDCG, além de explorar alternativas de infraestrutura - até definir a Vast.ai como solução mais eficiente para os experimentos. Com essa base, iniciei uma série de *baselines* utilizando *datasets* em inglês, português e também de domínio específico (artes), conduzindo experimentos tanto no Colab quanto em GPUs alugadas. Em seguida, selecionei o artigo "*ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval*"[14] como foco de reprodução, por apresentar uma metodologia clara e suficientemente detalhada, mesmo sem código público disponível. Implementei sua abordagem de *re-ranking*, adaptei o *pipeline* para conjuntos menores e executei treinamentos em GPUs, registrando cuidadosamente cada etapa (**Apêndice 5**). Paralelamente, iniciei também um *fine-tuning* do SigLIP[15] para fins comparativos, ampliando a análise sobre o comportamento dos modelos em diferentes cenários. Os códigos desenvolvidos, os resultados preliminares dos *baselines* e os detalhes da implementação da arquitetura de *re-ranking* (ELIP) estão documentados no **Apêndice 5**.

Na **Semana 10**, concluí minha Jornada consolidando tanto a compreensão teórica quanto a prática em Modelos Multimodais, com foco específico na relação texto–imagem, integrando tudo o que estudei e experimentei ao longo do processo. Aprofundei a implementação do ELIP, analisando com clareza seus desafios reais, como a necessidade de *datasets* massivos que ultrapassam centenas de gigabytes, e, ainda assim, repliquei a metodologia do artigo por meio do desenvolvimento do meu próprio código e da execução de experimentos com conjuntos menores. Paralelamente, refinei meu mapa mental, ampliei os *baselines* e conduzi testes sistemáticos com modelos CLIP e SigLIP em diferentes domínios e idiomas - o que me permitiu comparar diretamente o desempenho das arquiteturas e observar como respondem a adaptações de domínio. Os experimentos de *fine-tuning* revelaram ganhos concretos em métricas como Recall@K e nDCG, especialmente em cenários de nicho específicos. Além disso, calculei o custo por

[14] Zhan, Guanqi, et al. "ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval." *arXiv preprint arXiv:2502.15682* (2025).

[15] Zhai, Xiaohua, et al. "Sigmoid loss for language image pre-training." *Proceedings of the IEEE/CVF international conference on computer vision*. 2023.

experimento, considerando a GPU utilizada, tempo de execução e tamanho do *dataset* - o que me permitiu analisar de forma criteriosa o custo-benefício de cada abordagem e identificar quais estratégias eram mais eficientes para combinar desempenho e recursos computacionais. Também realizei uma série de treinamentos e inferências adicionais, documentando tempo, custo e resultados, que variaram conforme o volume de dados e a natureza de cada domínio. Paralelamente, estruturei um repositório completo com tutoriais, explicações, códigos e demonstrações, incluindo aplicações utilizando a biblioteca Gradio para exibir a avaliação de métricas, a busca por descrição e um *pipeline* de RAG multimodal - transformando, assim, o conhecimento adquirido em ferramentas reutilizáveis e acessíveis. Todos os detalhes de implementação de arquiteturas, treinamentos, materiais, datasets, avaliações e demonstrações estão no **Apêndice 6**.

Encerrando o ciclo, percebi que, mais do que dominar conceitos e reproduzir experimentos, desenvolvi autonomia para planejar, avaliar cenários, recalibrar escolhas e construir soluções multimodais de forma crítica e responsável. Nesse fechamento da minha Jornada, compreendi que o aprendizado não se restringiu aos Modelos Multimodais, mas envolveu também a capacidade de estruturar projetos complexos, dimensionar recursos, lidar com restrições reais e conduzir investigações de modo independente, consolidando a formação construída ao longo de todo período.

Registro meus agradecimentos a todos os professores excepcionais que tive a honra de conhecer ao longo desta jornada. À minha mãe, pelo apoio incondicional; à minha família e aos amigos, pela presença constante; e ao Daniel e ao Bernardo, que fizeram dias comuns serem extraordinários.

APÊNDICE 1

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 2 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Decisão de tema: Defini que meu tema de estudo será Modelos Multimodais, com foco específico na relação Texto–Imagem.

Modelos Multimodais Texto - Imagem é uma área ampla, resolvi categorizar a área para melhor organização.
Usei repositórios de surveys no github, que me ajudaram a ter uma visão mais ampla e categorizada. Respositórios Subdivisões de Multimodal Models

Curadoria de papers:

- Listei os trabalhos com os quais já tive contato direto ou indireto (CLIP, SigLIP 1 e 2, Flamingo, SmolVLM).
- Pedi apoio ao GPT para organizar uma lista mais completa, cobrindo desde os primeiros até os mais recentes papers.
- Adicionei referências vindas de repositórios GitHub e consolidei tudo em uma lista única para estudo.
- Explorei também o Research Rabbit como ferramenta de apoio.
- Papers - multimodal

Iniciei os estudos.

- Li o paper do modelo CLIP e registrei anotações: CLIP

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Anotações e sínteses:
    - Produzir resumos em formato de tópicos para cada leitura.
    - Criar fichamentos destacando diferenças metodológicas entre os modelos.
- Exploração de ferramentas:
    - Aprender melhor os recursos do Research Rabbit para mapear coautorias e evolução temporal.
    - Identificar outras plataformas úteis para visualização de citações e organização (ex.: Connected Papers, Semantic Scholar).
    - Explorar o Consensus.
- Se houver necessidade, alterar ou atualizar a lista de papers para estudo.
- Construção de base teórica:
    - Revisitar os principais conceitos de aprendizado contrastivo no geral, que servem como base para CLIP e modelos relacionados.
    - Se houver necessidade, revisitar outros conceitos gerais.
    - Começar a mapear benchmarks multimodais mais usados (ImageNet, COCO, LAION, etc.) e relacioná-los aos papers.
    - Estudar esses benchmarks como tipo de dados, quantidade, etc.
- Planejamento futuro:
    - Começar a pensar em possíveis perguntas de pesquisa relacionadas a texto–imagem.

Observação: **[caso precise fazer alguma observação, de qualquer "natureza"]**

O processo não foi linear, iniciei estudando alguns papers, depois fiz uma lista, então percebi que a categorização estava confusa para mim. Então dei um passo para trás e resolvi fazer essa

“categorização” e então reorganizar minha lista de artigos.

Observação: **[caso precise fazer alguma observação, de qualquer “natureza”]**

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Repositórios - Surveys Base para o Início do Estudo

Repositórios - Surveys que me auxiliaram no início do processo de conhecimento e mapeamento da área.

https://github.com/pliang279/awesome-multimodal-ml
https://github.com/donghao51/Awesome-Multimodal-Adaptation
https://github.com/friedrichor/Awesome-Multimodal-Papers
https://github.com/YingqingHe/Awesome-LLMs-meet-Multimodal-Generation
https://github.com/AIDC-AI/Awesome-Unified-Multimodal-Models
https://github.com/opendilab/awesome-multi-modal-reinforcement-learning
https://github.com/ys-zong/awesome-self-supervised-multimodal-learning

Subdivisões e Categorizações de Modelos Multimodais

Fundamentos do Aprendizado Multimodal

- Representação (como representar cada modalidade e como uni-las)
- Alinhamento (mapear diferentes modalidades para um mesmo espaço)
- Fusão (combinar informações de modalidades: early, late, híbrido, attention)
- Pré-treinamento Multimodal (contrastivo, generativo, auto-supervisionado)

Arquiteturas

- Modelos Encoders Multimodais (ex.: CLIP, SigLIP, ALBEF)
- Modelos Decoders Multimodais (ex.: DALL·E, Imagen)
- Modelos Encoder–Decoder Multimodais (tradução entre modalidades)

- MLLMs (Multimodal Large Language Models)
    - Baseados em LLMs + adaptadores (ex.: Flamingo, BLIP-2, LLaVA)
    - Unificados (um só backbone para várias modalidades, ex.: Kosmos, Gemini)

Tarefas Centrais

- Compreensão Multimodal
    - VQA, captioning, grounding, retrieval, commonsense reasoning
- Geração Multimodal
    - Texto → Imagem, Texto → Vídeo, Texto → Áudio, Texto → 3D
- Tradução Multimodal
    - Ex.: fala → texto, voz → face, imagem → fala
- Interação Multimodal
    - Diálogo multimodal, agentes, navegação, embodied AI

Outros

- Benchmarks
- Edição

Lista Completa de Papers – Subdivisões de Multimodal Models

Trabalhos selecionados de forma categorizada e cronológica.

Fundamentos

Paper	Ano	Link	Subdivisão
DeViSE	2013	https://arxiv.org/abs/1312.5650	Representação & Alinhamento
VirTex	2020	https://arxiv.org/abs/2006.06666	Pré-treinamento multimodal supervisionado
ALIGN (Google)	2021	https://arxiv.org/abs/2102.05918	Pré-treinamento multimodal em larga escala
LiT (Google)	2021	https://arxiv.org/abs/2111.07991	Pré-treinamento eficiente
Survey – Data Augmentation multimodal	2025	https://arxiv.org/abs/2501.18648	Pré-treinamento + Augmentation
MMRL	2025	https://arxiv.org/abs/2503.08497	Representação multimodal
CoMP	2025	https://arxiv.org/abs/2503.18931	Continual pretraining multimodal

Arquiteturas – Encoders

Paper	Ano	Link	Subdivisão
CLIP (OpenAI)	2021	https://arxiv.org/abs/2103.00020	Encoder multimodal
Florence (Microsoft)	2021	https://arxiv.org/abs/2111.11432	Foundation model multimodal
BLIP (Salesforce)	2022	https://arxiv.org/abs/2201.12086	Encoder multimodal (base para MLLMs)
SigLIP (Google)	2023	https://arxiv.org/abs/2303.15343	Encoder multimodal eficiente
SigLIP 2 (Google DeepMind)	2025	https://arxiv.org/abs/2502.14786	Encoder multimodal multilingue + dense features

Arquiteturas – Decoders

Paper	Ano	Link	Subdivisão
DALL·E (OpenAI)	2021	https://arxiv.org/abs/2102.12092	Decoder multimodal (Texto → Imagem)
Imagen (Google)	2022	https://arxiv.org/abs/2205.11487	Decoder multimodal (Texto → Imagem)
Parti (Google)	2022	https://arxiv.org/abs/2206.10789	Decoder multimodal autoregressivo (Texto → Imagem)

Arquiteturas – Encoder–Decoder

Paper	Ano	Link	Subdivisão
FLAVA (Meta)	2022	https://arxiv.org/abs/2112.04482	Encoder–Decoder multimodal

SimVLM (Google)	2022	https://arxiv.org/abs/2108.10904	Encoder–Decoder multimodal

Arquiteturas – MLLMs

Paper	Ano	Link	Subdivisão
Flamingo (DeepMind)	2022	https://arxiv.org/abs/2204.14198	MLLM (LLM + adaptadores)
Kosmos-1 (Microsoft)	2023	https://arxiv.org/abs/2302.14045	MLLM unificado
LLaVA	2023	https://arxiv.org/abs/2304.08485	MLLM open-source
SmolVLM	2025	https://arxiv.org/abs/2504.05299	MLLMs compactos
MMaDA	2025	https://arxiv.org/abs/2505.15809	Modelo generativo multimodal (Diffusion + RL)
Survey – Generative Categories in MLLMs	2025	https://arxiv.org/abs/2506.10016	Survey sobre arquiteturas gerativas em MLLMs

Desafios Transversais

Paper	Ano	Link	Subdivisão
The Multimodal Paradox	2025	https://arxiv.org/abs/2505.03020	Robustez & Equidade

Benchmarks

Paper	Ano	Link	Subdivisão
MMKC-Bench	2025	https://arxiv.org/abs/2505.19509	Benchmark de conflitos multimodais

Surveys Gerais

Paper	Ano	Link	Subdivisão
Unified Multimodal Understanding and Generation Models	2025	https://arxiv.org/abs/2505.02567	Survey unificado (Fundamentos + Arquiteturas)

Anotações sobre o Paper CLIP - Learning Transferable Visual Models From Natural Language Supervision

Contexto

- Antes do CLIP, os modelos de visão eram treinados em datasets como ImageNet, restritos a um conjunto fixo de rótulos.
- Modelos de linguagem, como BERT, eram treinados em grandes corpora textuais.
- Não havia um mecanismo direto que permitisse alinhar imagens e textos em um mesmo espaço semântico.
- Os datasets multimodais existentes, como MSCOCO, eram limitados em tamanho (centenas de milhares de exemplos), o que restringia a generalização.

Ideia central

- O CLIP propõe o uso de pares imagem+texto extraídos da web em larga escala para treinar dois encoders independentes (um de imagem e outro de texto) que projetam suas entradas em um espaço vetorial compartilhado.
- O objetivo é que imagens e textos semanticamente correspondentes ocupem regiões próximas desse espaço.

Dataset

- O modelo foi treinado com aproximadamente 400 milhões de pares (imagem, legenda) coletados da internet.
- Essa escala é duas ordens de magnitude maior do que datasets como ImageNet (~1,3 milhão de imagens) e permite cobrir uma ampla gama de conceitos visuais e linguísticos sem curadoria manual.

Arquitetura

O CLIP é composto por duas torres:

- Encoder de texto: baseado em Transformer, similar ao BERT, com embeddings finais normalizados em uma dimensão comum.
- Encoder de imagem: implementado com ResNet modificada ou Vision Transformer (ViT). No caso do ViT, a imagem é dividida em patches que passam por camadas de autoatenção, resultando em um embedding global.

Ambos os encoders são projetados para produzir embeddings de mesma dimensão (ex.: 512), permitindo cálculo direto de similaridade.

Treinamento contrastivo

- O processo de treinamento utiliza batches com N pares (imagem, texto). Cada encoder gera N embeddings.

- Calcula-se a similaridade do cosseno entre todos os pares possíveis, produzindo uma matriz N×N.
- O par correto deve ter similaridade máxima, enquanto os pares incorretos devem ser penalizados.
- A função de perda é uma versão simétrica da InfoNCE loss:
    - Para cada imagem, aplica-se softmax sobre as similaridades com todos os textos.
    - Para cada texto, aplica-se softmax sobre as similaridades com todas as imagens.
    - A perda é a média das entropias cruzadas das duas direções.
- Esse esquema força alinhamento bidirecional e garante que cada embedding seja discriminativo em relação a todos os outros do batch.

Representações aprendidas

O CLIP aprende um espaço semântico multimodal no qual:

- Imagens de gatos e a frase "um gato" ficam próximas.
- Diferentes estilos (“pintura impressionista”, “foto em preto e branco”) também se alinham.
- O modelo generaliza para conceitos complexos que não apareceram explicitamente no treinamento, pois aprende de distribuições amplas da web.

Zero-shot learning

A principal inovação prática é a capacidade de classificação zero-shot. Para classificar imagens em um novo dataset:

- Define-se as classes em forma textual, como “uma foto de um cachorro” ou “uma foto de um gato”.
- O encoder de texto gera embeddings para cada descrição.
- A imagem é passada pelo encoder de imagem.
- A classe é escolhida pelo texto com embedding mais próximo da imagem.

Esse procedimento substitui o treinamento supervisionado convencional e mostrou desempenho competitivo com modelos ajustados em cada dataset específico.

Impacto

- O CLIP demonstrou que pré-treinamento multimodal em larga escala com dados ruidosos da web pode superar abordagens supervisionadas baseadas em datasets curados.

- O trabalho inaugurou uma linha de pesquisa em modelos multimodais que hoje são padrão em visão+linguagem.

Pipeline resumido

1. Preparação do dataset com pares imagem+texto.
2. Codificação de imagens pelo encoder de visão.
3. Codificação de textos pelo encoder de linguagem.
4. Normalização dos embeddings.
5. Cálculo da matriz de similaridade imagem–texto.
6. Aplicação da InfoNCE loss simétrica.
7. Treinamento até que os pares corretos sejam consistentemente alinhados no espaço vetorial.

Aplicações

- Classificação zero-shot em múltiplos benchmarks de visão.
- Recuperação multimodal: busca de imagens a partir de texto e de textos a partir de imagens.
- Filtragem de conteúdo em pipelines de geração.
- Uso como avaliador em modelos generativos (por exemplo, CLIPScore).
- Base para arquiteturas posteriores como DALL·E, Stable Diffusion, BLIP, SigLIP e Flamingo.

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 10 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

- Continuação dos estudos dos papers e anotações em tópicos:
  - ALIGN: ALIGN
  - SigLIP: SigLIP
  - SigLIP2: SigLIP2
- Comparativo entre os papers estudados:
  - Registro de: Modelo, Ano, Organização, Arquitetura, Dados de Treino (tipo), Loss, Diferenças-chave.
  - Comparativo_papers
- Observação, pesquisa e análise dos benchmarks utilizados nos papers estudados.
  - Registro de Nome, Paper do Benchmark, Link disponível, em quais papers foi utilizado e uma breve descrição (característica dos dados, tarefa ou finalidade).
  - Organização em forma de tabela - com os campos citados acima: Benchmarks_multimodais
  - Exemplo de Benchmark registrado:
    - ImageNet, https://arxiv.org/abs/1409.0575, https://image-net.org/challenges/LSVRC, CLIP, ALIGN, SigLIP, SigLIP 2, Benchmark principal de classificação de imagens (1000 classes). Base zero-shot.

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

- Estudar mais papers - principalmente mais recentes.

- Testar algum deles na GPU ou Google Colab.
- Continuar atualizando as anotações (tópicos, comparativos e tabela de benchmarks)
- Pesquisar como são dados de pré-treino e de fine tuning dos modelos - principalmente os mais recentes.

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Anotações sobre o Paper ALIGN - Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision

Abstract

- Problema: Representações visuais ainda dependem de datasets curados e caros.
- Proposta: Usar 1B+ pares imagem-alt-text ruidosos, sem curadoria pesada.
- Modelo: Dual-encoder simples + contrastive loss (ALIGN).
- Resultados: Zero-shot classification no ImageNet. SOTA em Flickr30K e MSCOCO (retrieval). Cross-modal search (texto, imagem, imagem+texto).

1. Introduction

- NLP já se beneficia de grandes corpora não anotados (BERT, GPT, T5).
- Visão e visão-linguagem ainda usam datasets menores, curados (ImageNet, CC, COCO).
- Problema: custo de curadoria impede escalar.
- Solução: usar dataset muito maior, mas ruidoso.
- Metodologia: dual-encoder + contraste → embeddings compartilhados.
- Contribuição: provar que escala compensa ruído.

2. Related Work

- Visual representation learning: supervisionado (ImageNet, JFT), auto-supervisionado (SimCLR, MoCo).
- Visão + linguagem: aprendizado com captions, mas datasets pequenos (Flickr, COCO).
- Modelos com atenção cruzada → bons, mas lentos e caros.

- CLIP (OpenAI, 2021): semelhante, mas com dataset curado.
- Diferença-chave: ALIGN usa dados crus da web, sem lista pré-definida de conceitos.

3. Large-Scale Noisy Dataset

- Construção baseada no Conceptual Captions, mas sem limpeza pesada.
- Escala: 1,8B pares imagem+texto.
- Filtros aplicados: remover pornografia, baixa resolução, duplicatas, textos curtos/longos demais ou raros.
- Resultado: dataset enorme e ruidoso, mas adequado para escalabilidade.

4. Pre-training and Task Transfer

- Arquitetura: EfficientNet (imagem) + BERT (texto).
- Loss: contraste bidirecional (imagem→texto, texto→imagem).
- Transferência: retrieval (Flickr30K, MSCOCO, CxC), visual classification (ImageNet e variantes), fine-grained datasets e VTAB.

5. Experiments and Results

- Treino: 1024 TPUs v3, batch global 16k, otimizador LAMB, 12 épocas.
- Resultados:

    1. Retrieval → supera CLIP e modelos de atenção.
    2. Zero-shot classification (ImageNet) → 76.4% top-1.
    3. Visual classification (fine-tuning) → 88.64% top-1 no ImageNet, robusto em variantes.
    4. VTAB → 79.99%, melhor que BiT-L.
    5. Fine-grained tasks → comparável a SOTA (Flowers, Pets, Cars, Food101).

6. Ablation Study

- Arquiteturas: eficiência cresce com backbones maiores.
- Dimensão de embedding: maior → melhor.
- Batch negatives: reduzir degrada performance.
- Temperatura: aprendida automaticamente → bom resultado.
- Dataset size: mais dados ruidosos superam dados limpos pequenos.

7. Analysis of Embeddings

- Consultas multimodais: imagem+texto → busca por conceitos compostos.
- Composicionalidade: adição/subtração de embeddings altera atributos (ex.: cor).
- Generalização: encontra landmarks e obras mesmo não vistos no treino.

8. Multilingual ALIGN

- Dataset expandido para 100+ idiomas (1,8B pares).
- Novo vocabulário de 250k.
- Avaliação: Multi30K (retrieval em inglês, alemão, francês, tcheco).
- Resultados: zero-shot > M3P em todas as línguas. Comparable ao UC2 mesmo sem fine-tuning.

9. Conclusion

- Método simples + dataset enorme = representações multimodais fortes.
- Escala supera qualidade de curadoria.
- Resultados SOTA em retrieval, classificação e transferência.

10. Social Impacts & Future Work

- Riscos: alt-texts podem conter conteúdo tóxico ou enviesado.
- Pode reforçar estereótipos culturais e sociais.
- Potencial de uso indevido em vigilância ou propaganda.
- Futuro: balanceamento, fairness, testes culturais e religiosos.
- Uso responsável é essencial.

Anotações sobre o Paper SigLip - Sigmoid Loss for Language Image Pre-Training

Abstract

- Proposta de uma loss alternativa para pré-treinamento visão-linguagem: Sigmoid Loss.
- Diferente da loss contrastiva com softmax (InfoNCE), não exige normalização global por batch.
- Vantagens:
    - Melhor em batches pequenos (<16k).
    - Permite batches muito maiores (até 1M).
    - Mais eficiente em memória e simples de implementar.
- SigLiT (LiT + Sigmoid) treinado com apenas 4 TPUs alcança 84,5% de acurácia zero-shot no ImageNet em 2 dias.
- Batch size de 32k já é suficiente, ganhos adicionais saturam.

1. Introduction

- Pré-treinamento contrastivo em pares imagem-texto se tornou padrão após CLIP e ALIGN.
- Normalmente usa-se loss com softmax → requer operações em todo o batch, caro e limitado.
- Sigmoid Loss simplifica e permite escalar sem gargalos de memória.
- Introduz dois frameworks:
    - SigLiT (Locked-image Tuning + Sigmoid).
    - SigLIP (CLIP-like, do zero com Sigmoid).
- Objetivo: tornar o pré-treinamento multimodal mais acessível e eficiente.

2. Related Work

- Contrastive learning: InfoNCE (softmax) domina, mas sigmoid já foi usado em classificação supervisionada.
- Vision-language pre-training: CLIP, ALIGN → resultados fortes, mas exigem grandes batches.
- Abordagens alternativas: modelos generativos (SimVLM, GIT, CoCa, BLIP).
- Trabalhos de eficiência: LiT, FLIP, BASIC, LAION.
- Gap: poucos estudaram eficiência da loss, foco maior em escalar dados e modelos.

3. Method

- Revisão da loss contrastiva com softmax: normalização bidirecional (imagem→texto, texto→imagem).
- Problema: precisa de matriz |B|x|B| de similaridades → caro em memória.
- Proposta: Sigmoid Loss trata cada par individualmente (positivos e negativos), formulada como classificação binária.

- Introduz viés treinável (bias b) para corrigir o desequilíbrio entre muitos negativos e poucos positivos.
- Implementação eficiente em “chunks”: cada dispositivo processa apenas parte dos pares, reduzindo memória de $|B|^2$ para b^2.
- Permite batch sizes extremamente grandes (até 1M).

4. Results

- Avaliação em zero-shot ImageNet e retrieval no XM3600 (36 línguas).
- Principais achados:
    - SigLiT: melhor em batches pequenos; saturação em ~32k; treinou até 1M batch.
    - SigLIP: superou CLIP no regime de batch <32k; mais eficiente em memória.
    - mSigLIP: batch de 32k já é ótimo para setup multilíngue; escalas maiores degradam.
    - Poucos recursos: com apenas 4 TPUs, SigLiT chega a 84,5% ImageNet zero-shot em 2 dias.
    - Fine-tuning SigLIP: ao usar pesos pré-treinados, desativar weight decay nos encoders melhora estabilidade.
    - Scaling up: versões maiores (ViT-L, So400M) superam CLIP, OpenCLIP, EVA-CLIP em classificação e retrieval.
    - Estabilização: reduzir β2 do Adam/AdaFactor (0.999 → 0.95) evita instabilidade em batches grandes.
    - Ratio positivos/negativos: não crítico, mas hard negatives ajudam mais que easy negatives.
    - Bias term: inicializar b = -10 evita overshoot inicial e melhora resultados.

- Robustez a ruído: Sigmoid Loss mostra maior resistência a imagens/textos corrompidos e alinhamentos trocados.

5. Conclusion

- Sigmoid Loss é simples, eficiente e mais robusta que Softmax em batch pequeno.
- Permite explorar batches gigantes sem custo desproporcional de memória.
- Batch size de 32k é suficiente, escalas maiores pouco contribuem.
- Introduz estabilidade, eficiência e robustez importantes para pré-treinamento visão-linguagem.
- Espera-se que facilite pesquisas multimodais com menos recursos.

Anotações sobre o Paper SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features

Abstract

- Introdução do SigLIP 2, nova geração de encoders visão-linguagem multilíngues.
- Combina treinamento contrastivo do SigLIP com:
  - Pré-treinamento baseado em captioning (LocCa).
  - Losses auto-supervisionados (self-distillation, masked prediction).
  - Curadoria online de dados.
- Avanços:
  - Melhor zero-shot classification, retrieval e transferência para VLMs.

    - Melhoria em tarefas de localização e predições densas (segmentação, profundidade).
    - Suporte a múltiplas resoluções e preservação do aspecto nativo (NaFlex).
    - Dataset mais diverso, com técnicas de de-biasing para fairness.
- Modelos lançados em 4 tamanhos: ViT-B (86M), L (303M), So400m (400M) e g (1B).

1. Introduction

- CLIP e ALIGN popularizaram treinamento contrastivo em larga escala.
- Permite zero-shot classification, retrieval eficiente e integração em VLMs.
- Avanços recentes: recaptioning, auto-supervisão, uso de decoders.
- Problema: releases open-source ainda seguem CLIP clássico, sem integrar todas as melhorias.
- Contribuição do SigLIP 2: unificar técnicas em uma receita moderna → ganhos em tasks densas, localização e robustez.

2. Training recipe

- Base: SigLIP com sigmoid loss.
- Acréscimos:
    - Decoder-based pretraining (LocCa) para captioning, grounded captioning e referring expressions.
    - Self-distillation + masked prediction (SILC, TIPS) para features locais/densas.
    - Estratégia em etapas (staged training) para gerenciar custo computacional.

2.1. Arquitetura, dados, otimizador

- Arquitetura mantém compatibilidade com SigLIP (ViT + MAP pooling).

- Texto: tokenizer Gemma multilíngue (256k vocabulário, lowercase).
- Dados: WebLI (10B imagens, 12B textos, 109 línguas).
    - Mistura 90% inglês / 10% não-inglês.
    - Filtros de bias aplicados.
- Treino: Adam lr 1e-3, batch 32k, 40B exemplos, até 2048 TPUs v5e (FSDP).

2.2. Treinamento com Sigmoid + decoder

- Sigmoid loss: classificação binária de pares (match / não-match).
- Decoder: cross-attention com visão → aprende captioning, grounded captioning e bounding boxes.
- Caption tokens preditos em paralelo (não causais).
- Decoder só usado no treino, não incluído nos checkpoints.

2.3. Self-distillation e masked prediction

- Self-distillation: correspondência local→global (teacher EMA).
- Masked prediction: patches mascarados → student prevê features do teacher.
- Ativados apenas nos 20% finais do treino.
- Pesos ajustados para balancear tarefas globais vs locais.

2.4. Adaptação a diferentes resoluções

- Fixed-resolution: retreino no final para múltiplas resoluções.
- NaFlex: preserva aspecto nativo + suporta várias resoluções numa só checkpoint.

- Inspirado em FlexiViT e NaViT.

2.5. Distillation via active data curation

- Para modelos pequenos (B/16, B/32).
- Usa ACID (Active Data Curation via implicit distillation).
- Teacher forte (SigLIP 2 So400m) seleciona batches com exemplos mais úteis.
- Melhora performance sem custo extra de compute explícito de distillation.

3. Experiments and results
   3.1. Zero-shot classification e retrieval

- SigLIP 2 supera SigLIP e outros open baselines (OpenCLIP, MetaCLIP, EVA-CLIP, DFN).
- Melhora mais acentuada em modelos pequenos (B).
- Multilingual retrieval (XM3600, 36 línguas): SigLIP 2 próximo do mSigLIP, mas muito melhor em inglês.

3.1.1. NaFlex variant

- Avaliação em benchmarks OCR/document (TextCaps, HierText, SciCap, Screen2Words).
- NaFlex > versão padrão em tasks com distorção de aspecto.
- Standard ainda melhor em natural images (graças à distillation extra).

3.2. SigLIP 2 como encoder para VLMs

- Integrado ao Gemma 2 LLM (PaliGemma recipe).
- Melhor que SigLIP e AIMv2 em VQA, captioning, OCR, grounded captioning etc.

3.3. Dense prediction tasks

- Segmentation (ADE20k, Pascal), depth (NYUv2), surface normals.
- SigLIP 2 > SigLIP e CLIP em todas, com ganhos grandes.
- Open-vocabulary segmentation (Cat-Seg): SigLIP 2 L/16 supera até OpenCLIP G/14.

3.4. Localization tasks

- Referring expression comprehension (RefCOCO, RefCOCO+, RefCOCOg):
  - SigLIP 2 >> SigLIP e CLIP.
  - Só perde para LocCa (que usa captions só em inglês).
- Open-vocabulary detection (OWL-ViT adaptado):
  - SigLIP 2 > SigLIP em COCO e LVIS, principalmente em categorias raras.

3.5. Cultural diversity e fairness

- Dados multilíngues (10%) + filtros de de-biasing.
- Avaliação em Dollar Street, GeoDE, GLDv2 → melhor geodiversidade e fairness.
- Representation bias (gênero-ocupação) cai de 35% (SigLIP) para ~7% (SigLIP 2).
- Disparidades por renda/região também reduzem, mas de forma mais sutil.

4. Related Work

- CLIP, ALIGN → fundação contrastiva.
- Trabalhos open-weight: OpenCLIP, MetaCLIP, DFN, EVA-CLIP, SigLIP.
- Extensões: recaptioning, losses adicionais, decoders, self-supervisão.
- SigLIP 2 é o primeiro a unificar múltiplas melhorias em uma só receita.

5. Conclusion

- SigLIP 2 = evolução significativa dos encoders CLIP-like.
- Melhora zero-shot, retrieval, dense tasks e localização.
- Mais justo culturalmente, menos enviesado.
- NaFlex adiciona suporte multi-resolução + aspecto preservado.
- Lançado open-weight em 4 tamanhos para trade-off custo/performance.

Comparativo entre Papers

Modelo	Ano	Organização	Arquitetura	Dados de Treino	Loss	Diferenças-Chave
CLIP	2021	OpenAI	Dual-encoder (texto + imagem com Transformer + ResNet/ViT)	400M pares image m-texto (web)	Contrastive loss (InfoNCE)	Primeiro grande modelo multimodal escalável; forte generalizaç ão zero-shot.
ALIGN	2021	Google	Dual-encoder (similar ao CLIP)	Bilhões de pares image m-texto ruidoso s (web sem filtrage m	Contrastive loss	Mostrou que escalar dados ruidosos gera ganhos expressivos.

				pesada)		
SigLIP	2024	Google DeepMind	Encoder de imagem + encoder de texto	Centenas de milhões de pares image m-texto	Sigmoid loss (independente por par)	Substitui a softmax global pela sigmoid; mais robustez e consistência.
SigLIP 2	2025	Google DeepMind	Similar ao SigLIP, otimizado para multilinguismo e localização	Pares em múltiplos idiomas + dados de grounding espacial	Sigmoid loss (aprimorada)	Suporte a múltiplos idiomas, grounding espacial (bounding boxes) e dense retrieval.

Benchmarks Levantados

Registro de Benchmarks - Mapeamento de benchmarks utilizados

Nome	Paper	Link	Papers	Descrição
ImageNet	arXiv:1409.0575	image-net.org	CLIP, ALIGN, SigLIP, SigLIP 2	Benchmark principal de classificação de imagens (1000 classes). Base zero-shot.

ImageNet-v2	MLR 2019	GitHub	SigLIP	Nova versão do ImageNet para avaliar robustez.
ImageNet ReaL	arXiv:2006.09829	GitHub	SigLIP, SigLIP 2	Reanotação com múltiplos rótulos por imagem.
ObjectNet	NeurIPS 2019	objectnet.dev	SigLIP, SigLIP 2	Testa robustez a mudanças de contexto, ângulo e fundo.
MS-COCO (Retrieval)	arXiv:1405.0312	cocodataset.org	CLIP, ALIGN, SigLIP	Recuperação imagem ↔ texto (image-to-text e text-to-image).
Flickr30k	ACL 2014	DenotationGraph	CLIP, ALIGN	Retrieval imagem ↔ legenda curta (30k imagens, 5 descrições cada).
CIFAR-10	Página oficial	Dataset	CLIP, ALIGN	Classificação de imagens pequenas em 10 classes.
CIFAR-100	Página oficial	Dataset	CLIP, ALIGN	Versão mais difícil do CIFAR, com 100 classes.
Oxford Pets	Dataset	Dataset	CLIP, ALIGN	Reconhecimento de raças de cães e gatos.
Oxford Flowers-102	arXiv:0811.4442	Dataset	CLIP, ALIGN	Classificação de flores em 102 categorias.
Caltech-101	Tech Report	Dataset	CLIP	Classificação de objetos em 101 categorias.
Stanford Cars	Dataset	Dataset	CLIP, ALIGN	Reconhecimento de modelos de carros.

Food-101	arXiv:1411.7923	Dataset	CLIP, ALIGN	Classificação de pratos de comida (101 classes).
SUN397	Projeto	Dataset	CLIP	Classificação de cenas em 397 categorias.
DTD (Textures)	Dataset	Dataset	CLIP	Reconhecimento de texturas visuais.
EuroSAT	arXiv:1709.00029	GitHub	CLIP	Classificação de imagens de satélite.
UCF101	arXiv:1212.0402	Dataset	CLIP	Reconhecimento de ações em vídeos curtos.
Kinetics-700	arXiv:1907.06987	DeepMind	CLIP	Benchmark de ações humanas em vídeo (700 classes).
iNaturalist	arXiv:1707.06642	GitHub	ALIGN	Reconhecimento de espécies (long-tail).
PASCAL VOC	Dataset	Dataset	ALIGN	Detecção e segmentação (transfer learning).
ADE20K	arXiv:1608.05442	Dataset	ALIGN, SigLIP 2	Segmentação semântica de cenas complexas.
XM3600 / Crossmodal-3600	EMNLP 2021	GitHub	SigLIP, SigLIP 2	Retrieval imagem ↔ texto em 36 idiomas.
RefCOCO	arXiv:1608.02973	GitHub	SigLIP 2	Localização baseada em linguagem (referring expressions).
LVIS	arXiv:1908.03195	Dataset	SigLIP 2	Detecção open-vocabulary (categorias raras).

COCO (Dense / Segmentation)	arXiv:1405.0312	cocodataset.org	ALIGN, SigLIP 2	Segmentação semântica e dense prediction.
Dollar Street	arXiv:1909.01602	GitHub	SigLIP 2	Benchmark cultural e socioeconômico.
GeoDE	arXiv:2306.11311	GitHub	SigLIP 2	Reconhecimento geográfico (país/região).
GLDv2	arXiv:2004.01804	GitHub	SigLIP 2	Google Landmarks v2: identificação de marcos turísticos.

APÊNDICE 2

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 17 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

- Leitura, estudo e produção de resumos em tópicos dos papers:
  - DeViSE: A Deep Visual-Semantic Embedding Model (2013 - primeiro modelo a unir embeddings de linguagem (Word2Vec) e CNNs para projetar imagens e palavras no mesmo espaço, porém ainda "divide"em duas partes separadas - texto e imagem - e adiciona uma camada de projeção) DeViSE
  - VirTex (2020: Resnet + Transformers com multi-head attention sobre tokens e features da imagem). VirTex
  - Flamingo: a Visual Language Model for Few-Shot Learning (2022: integra LLM pré-treinado com módulos multimodais para aprendizado in-context em visão e linguagem) Flamingo
  - Language Is Not All You Need: Aligning Perception with Language Models (2023: MLLM (Multimodal Large Language Models) treinado do zero em corpora web multimodais) Kosmos 1

- Estudo de datasets de pré-treino e fine tuning. Datasets
  - Exemplo:
    - Nome do Dataset: **LAION-400M** —
    - Paper(s): **CLIP, ALIGN, Kosmos-1, SigLIP1, SigLIP2** —
    - Tipo de Dados: **Pares imagem-texto (web)** —
    - Quantidade de Dados: **400M pares** —

- Uso: **Pré-treino**

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Terminar minha tabela de leituras original - Linha de tempo multimodal.
- Ler Multimodal Large Language Models for Text-rich Image Understanding: A Comprehensive Review
- Nova lista de leituras - com um maior direcionamento.
- Continuação do estudo sobre datasets.

Observação: **[caso precise fazer alguma observação, de qualquer "natureza"]**

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Anotações sobre o Paper DeViSE (Deep Visual-Semantic Embeddings)

Ainda divide visão e language e depois projeta para a mesma. Word2vec + CNN. 2013.

Abstract

- Visual Recognition sofre limitações ao escalar para muitas categorias, devido à dificuldade de obter dados anotados.-> limitações de dataset
- A proposta é usar dados de texto não anotado como fonte semântica para treinar e restringir predições.
- O DeViSE combina imagens rotuladas e embeddings semânticos aprendidos de texto.
- O modelo alcança desempenho comparável ao estado da arte no ImageNet 1K.
- Erros cometidos são semanticamente mais razoáveis.
- Permite generalização para milhares de rótulos não vistos (zero-shot learning), com acertos significativos mesmo sem exemplos visuais.

Introduction

- O mundo visual contém um número vasto de objetos, com rótulos muitas vezes ambíguos ou múltiplos.
- Sistemas atuais de visão tratam a tarefa como classificação N-way, com categorias fixas e independentes.
- Esse formato artificial limita a escalabilidade e não aproveita relações semânticas entre rótulos.
- A proposta é respeitar a continuidade natural do espaço visual em vez de compartimentar categorias.
- O DeViSE integra informações semânticas de texto não anotado para enriquecer modelos de visão.
- O objetivo é criar representações visuais alinhadas ao espaço semântico, permitindo erros mais plausíveis e generalização zero-shot para categorias nunca vistas.

Previous Work

- O estado da arte era representado por CNNs profundas com softmax, que apresentam dificuldades em escalar para muitas classes.
- O WSABIE explorou embeddings conjuntos de imagem e texto, mas com mapeamento linear limitado e sem generalização para novas classes.
- Socher et al. treinaram CNNs com embeddings de palavras, mas em pequena escala (8 classes + 2 outliers), apresentando trade-offs entre desempenho em vistas e não vistas.

- Outros métodos de zero-shot dependiam de fontes manuais de semântica, como a hierarquia WordNet ou bases de atributos.
- O DeViSE difere ao aprender representações semânticas diretamente de texto não anotado, sem precisar de curadoria manual.

Proposed Approach

- O objetivo é transferir conhecimento semântico aprendido em texto para um modelo de reconhecimento visual.
- A abordagem envolve três etapas: pré-treino de um modelo de linguagem, pré-treino de um modelo visual, e a combinação em um modelo visual-semântico conjunto.

Language Model Pre-training

- Utiliza o skip-gram (Word2Vec) para aprender embeddings densos de palavras.
- Treinado em bilhões de palavras da Wikipedia, com vocabulário de 155k termos.
- Embeddings de 500 a 1000 dimensões capturam relações semânticas.
- Termos semanticamente próximos resultam em vetores próximos no espaço.

Visual Model Pre-training

- Baseado na CNN vencedora do ImageNet 2012 (AlexNet).
- Estrutura de convoluções, pooling, normalização, fully-connected e dropout.
- Treinado para 1000 classes no ImageNet 1K, reproduzindo resultados do AlexNet.

Deep Visual-Semantic Embedding Model - DeViSE

- Remove o softmax da CNN e adiciona uma camada de projeção linear para o espaço semântico. -> Adiciona uma camada linear que projeta o vetor visual (4096-D) no espaço semântico (500 ou 1000-D).
- Treinado para aproximar representações de imagem aos embeddings de texto dos rótulos.
- Treinado com hinge rank loss (maximizar similaridade com o embedding correto e minimizar com incorretos).
- Embeddings são normalizados e busca é feita por vizinhos mais próximos no espaço semântico.
- Avaliação feita por nearest neighbors no espaço semântico.

Results

ImageNet (ILSVRC) 2012 1K

- Comparação entre DeViSE, softmax baseline e embeddings aleatórios.
- Em métricas flat hit@k, DeViSE chega próximo ao softmax, especialmente para k maiores.

- Em métricas hierárquicas, DeViSE supera o softmax, mostrando erros semanticamente mais plausíveis.
- Embeddings aleatórios têm desempenho muito inferior, confirmando a importância da estrutura semântica aprendida.

Generalization and Zero-Shot Learning

- O DeViSE é capaz de predizer rótulos nunca vistos no treino de imagens, aproveitando semântica aprendida em texto.
- Avaliação em conjuntos zero-shot derivados do ImageNet 21K:
- 2-hop (categorias próximas das vistas).
- 3-hop (categorias mais distantes).
- Full 21K (todas as categorias não vistas).
- No zero-shot, o DeViSE alcança taxas de acerto significativas em top-k, chegando a mais de 36% em top-20 para classes 2-hop.
- O baseline softmax não consegue prever classes não vistas (0%).
- No hierarchical precision@k, o DeViSE supera o softmax em cenários mais difíceis, mostrando melhor uso da semântica.
- Comparado a métodos anteriores que usavam hierarquias manuais, o DeViSE obtém desempenho competitivo ou superior mesmo sem essa curadoria.

Conclusion

- O DeViSE atinge desempenho comparável a modelos de classificação com softmax no ImageNet 1K.
- Faz erros semanticamente mais razoáveis.
- Generaliza para milhares de categorias nunca vistas, mostrando capacidade de zero-shot learning em larga escala.
- Não depende de hierarquias manuais, apenas de texto não anotado.
- É compatível com datasets grandes e abertos, podendo escalar para aplicações práticas com milhares de categorias.
- O modelo abre caminho para sistemas que lidem com rótulos variáveis e sobrepostos.
- Perspectivas futuras incluem aproveitar ainda mais a estrutura dos embeddings de linguagem para maior escalabilidade e eficiência.

Anotações sobre o Paper VirTex: Learning Visual Representations from Textual Annotations

Resnet + Transformers com multi-head attention sobre tokens e features da imagem. 2020

Abstract

- A abordagem padrão em visão é pré-treinar em ImageNet com classificação supervisionada e transferir para outras tarefas.
- Métodos recentes exploram pré-treino não supervisionado em grandes quantidades de imagens sem rótulos.
- O VirTex busca aprender boas representações visuais a partir de menos imagens, revisitando o pré-treino supervisionado.
- Propõe usar anotações textuais densas (legendas) para treinar representações visuais.
- Treina CNNs do zero com COCO Captions e transfere para classificação, detecção e segmentação.
- VirTex produz representações que igualam ou superam ImageNet supervisionado ou não supervisionado, usando até 10× menos imagens.

Introduction

- O paradigma dominante é pré-treinar CNNs para classificação em ImageNet e depois transferir.
- Esse processo foi chave para avanços em detecção, segmentação, captioning e VQA.
- No entanto, é caro escalar, pois depende de imagens rotuladas manualmente.
- Métodos não supervisionados têm surgido como alternativa, chegando a milhões ou bilhões de imagens.
- A questão é se existem formas mais eficientes de usar menos imagens para aprender representações de alta qualidade.
- VirTex propõe pré-treino supervisionado alternativo, usando captions para fornecer sinais semânticos densos.
- Captions descrevem múltiplos objetos, atributos, relações e ações, fornecendo um aprendizado mais rico que rótulos de classificação.
- A coleta de captions é mais simples e barata do que construir ontologias de classes e rotular manualmente.
- O objetivo é mostrar que linguagem natural pode supervisionar representações visuais transferíveis de forma mais eficiente em dados.

Related Work

- Weakly Supervised Learning: usa grandes conjuntos de imagens da web com rótulos ruidosos (tags, hashtags). Aprende muito, mas com baixa qualidade de anotação.
- Self-Supervised Learning: pré-tarefas como previsão de contexto, colorização, rotação, inpainting, jigsaw, clustering e métodos contrastivos. Carecem de semântica, focam em pistas de baixo nível.

- Vision-and-Language Pretraining: modelos multimodais (VQA, raciocínio visual, retrieval) que geralmente dependem de CNN pré-treinada em ImageNet, detectores em Visual Genome e modelos de linguagem como BERT. São pipelines complexos e mantêm a visão como dependente do ImageNet.
- VirTex difere ao treinar do zero para captioning, e usar o backbone visual aprendido diretamente em tarefas visuais downstream.
- Trabalhos concorrentes também exploram captions, mas dependem de modelos de linguagem pré-treinados (ex. BERT) ou avaliam apenas em vídeo. VirTex treina tudo do zero e avalia em mais tarefas.

Method

- Objetivo: aprender representações visuais transferíveis a partir de pares imagem-caption.
- Captions trazem informação rica: presença de objetos, atributos, arranjos espaciais, ações.
- O pré-treino é feito com image captioning: dado uma imagem, prever tokens da legenda.
- Modelo tem dois componentes: visual backbone (CNN) e textual head (Transformers).
- O backbone extrai features da imagem; a textual head gera a legenda token a token.
- O modelo faz bicaptioning: dois decoders, um left-to-right e outro right-to-left.
- Após o pré-treino, descarta-se a textual head e só o backbone visual é transferido.

Visual Backbone

- ResNet-50 usado para comparação com baselines.
- Saída é um grid 7×7×2048 de features.
- Durante pré-treino, há uma projeção linear para facilitar atenção do decoder.

Textual Head

- Dois Transformers unidirecionais (forward e backward).
- Usam multi-head attention sobre tokens e features da imagem.
- Cada camada inclui self-attention mascarado, cross-attention com features visuais, feed-forward, normalização e residual.
- Tokenização via SentencePiece (BPE), vocabulário de 10k tokens.

Training

- Treinado em COCO Captions (118k imagens, 5 captions cada).
- Augmentations: crop, jitter de cor, normalização, flip horizontal (com troca de “left/right” na legenda).
- Otimização com SGD + momentum, weight decay, LookAhead, batch norm sincronizado, warmup + cosine decay.

- Backbone usa LR maior que textual head para acelerar convergência.
- Early stopping baseado em desempenho downstream no VOC07.

Experiments

Image Classification with Linear Models

- Avaliado em VOC07 (mAP) e ImageNet-1k (top-1 acc).
- VirTex supera métodos de auto-supervisão e rótulos alternativos em eficiência de anotação.
- Captions fornecem melhor custo-benefício que labels ou masks.
- VirTex é mais eficiente em dados: com 118k imagens, iguala ou supera ImageNet com 1.28M imagens.
- Melhor usar mais imagens com menos captions cada do que o contrário.

Ablations

- Tasks: bicaptioning supera forward captioning, token classification e MLM (MLM tem baixa eficiência amostral).
- Backbone: redes mais largas/profundas trazem ganhos, mas dependem de otimização.
- Transformer size: aumentar largura e profundidade melhora desempenho, até certo limite.

Fine-tuning Tasks for Transfer

- Avaliado em COCO e LVIS instance segmentation, VOC detection, iNaturalist fine-grained classification.
- VirTex supera métodos com número similar de imagens, e iguala ou excede ImageNet supervisionado e MoCo-IN, mesmo com 10× menos dados.
- Melhorias grandes em LVIS, mostrando eficácia em classes de cauda longa.

Image Captioning

- VirTex não visa SOTA em captioning, mas mostra desempenho modesto em CIDEr/SPICE.
- Atenções dos Transformers mostram foco em regiões corretas (objetos, fundos, ações).
- Demonstra que o modelo aprende features visuais semanticamente significativas.

Conclusion

- VirTex mostra que supervisionar visão com captions é competitivo com pré-treino supervisionado e auto-supervisionado em ImageNet.
- O foco é em tarefas visuais downstream, mas trabalhos futuros podem explorar transferência conjunta do backbone e textual head.
- O uso de captions abre caminho para escalar para dados de imagem-texto da web em larga escala, que são mais ruidosos, mas muito maiores que COCO.

Anotações sobre o Paper 🦩 Flamingo: a Visual Language Model for Few-Shot Learning

Abstract

- Propõe um modelo multimodal que aprende a partir de pares imagem–texto em larga escala.
- O Flamingo pode realizar tarefas de visão e linguagem em few-shot learning, com apenas alguns exemplos no prompt.
- Combina um grande language model pré-treinado com camadas de atenção multimodal para integrar imagens e vídeos.
- Mostra resultados fortes em tarefas de VQA, captioning, compreensão de vídeo, sem precisar de fine-tuning.
- Alcança desempenho SOTA em benchmarks multimodais de forma eficiente e flexível.

Introduction

- Modelos de linguagem em larga escala mostraram forte habilidade de few-shot learning em texto.
- Na visão computacional, modelos geralmente requerem fine-tuning supervisionado ou pré-tarefas específicas.
- Objetivo: trazer para o espaço multimodal a mesma flexibilidade que LLMs têm em texto.
- Flamingo é um Visual Language Model (VLM) que processa sequências de texto e imagens/vídeos de forma unificada.
- Treinado em larga escala em dados de imagem–texto e vídeo–texto, sem precisar de datasets de tarefas específicas.
- Suporta entrada multimodal intercalada (texto e imagens/vídeos alternados).

Related Work

- Vision-Language Pretraining (VLP): CLIP, ALIGN, SimVLM usam pares de imagem–texto, mas focam em embeddings ou predições específicas.

- VQA e captioning: modelos multimodais anteriores requerem treinamento supervisionado para cada tarefa.
- Large Language Models: GPT-3 demonstrou aprendizado in-context em texto, inspirando transferência dessa ideia para visão + linguagem.
- Flamingo integra os avanços de LLMs com VLP para criar um modelo multimodal few-shot geral.

Model Architecture

- O backbone é um Language Model pré-treinado de larga escala.
- Adiciona Perceiver Resampler modules, que transformam features visuais em embeddings compactos para interação com o LM.
- As entradas podem ser texto puro, texto com imagens, ou texto com vídeos (frames amostrados).
- O modelo gera texto de saída condicionado em ambas modalidades.
- Suporta prompts com múltiplos pares exemplo (few-shot) sem ajuste adicional.

Training

- Pré-treinado em um grande corpus multimodal de pares imagem–texto e vídeo–texto.
- Inclui datasets públicos como COCO Captions, Conceptual Captions (CC3M/CC12M), Visual Genome, VQAv2, OK-VQA, MSR-VTT, HowTo100M, WebVid-2M.
- Também usa um dataset proprietário web-scale chamado MassiveWeb.
- Otimização ajusta somente os módulos multimodais, mantendo o LM textual em grande parte congelado para preservar suas capacidades.
- A estratégia permite aproveitar o poder do LLM já treinado em texto.

Results

Few-shot and Zero-shot Performance

- Avaliado em benchmarks multimodais sem fine-tuning, apenas com exemplos no prompt.
- Vision tasks: VQAv2, OK-VQA, ScienceQA, HatefulMemes, visual reasoning benchmarks.

- Image captioning: COCO Captions, NoCaps.
- Video understanding: MSR-VTT, VATEX, HowTo100M.
- Flamingo supera modelos anteriores em zero-shot e few-shot, alcançando SOTA em diversas métricas.
- Demonstra forte generalização mesmo para tarefas não vistas no treino.

Ablation Studies

- O Perceiver Resampler é essencial para lidar com inputs visuais longos (como múltiplos frames de vídeo).
- O congelamento parcial do LM preserva a capacidade de few-shot em texto puro.
- A escala do pré-treino multimodal é crítica: mais dados → melhor performance.

Conclusion

- Flamingo é o primeiro modelo multimodal em larga escala a demonstrar aprendizado in-context few-shotsemelhante ao GPT-3, mas em visão + linguagem.
- Supera modelos anteriores em uma ampla gama de tarefas sem precisar de treino supervisionado por tarefa.
- Mostra a importância de integrar LLMs com módulos visuais flexíveis.
- Abre caminho para VLMs mais gerais, escaláveis e aplicáveis a diversos cenários multimodais.

Anotações sobre o Paper Kosmos - Language Is Not All You Need: Aligning Perception with Language Models

Abstract

- Introduz o Kosmos-1, um **Multimodal Large Language Model (MLLM)** capaz de perceber modalidades diversas, seguir instruções (zero-shot) e aprender em contexto (few-shot).
- Treinado do zero em corpora multimodais em escala web: texto puro, pares imagem–legenda e documentos intercalando texto e imagens.
- Avaliado em cenários zero-shot, few-shot e multimodal chain-of-thought prompting.

- Mostra desempenho em:
  - compreensão e geração de linguagem (incluindo OCR-free NLP a partir de imagens de documentos),
  - tarefas de percepção–linguagem (diálogo multimodal, captioning, VQA),
  - tarefas de visão (classificação de imagens guiada por descrições).
- Demonstra **transferência cross-modal** (de linguagem para multimodal e vice-versa).
- Introduz também um dataset de testes de QI Raven para avaliar raciocínio não verbal.

1 Introduction: From LLMs to MLLMs

- LLMs já funcionam como interface geral em NLP, mas não percebem nativamente modalidades como imagem ou áudio.
- A percepção multimodal é crucial para **AGI**, pois conecta modelos ao mundo real.
- Kosmos-1 busca alinhar percepção com LLMs, permitindo “ver e falar”.
- Segue a filosofia do **MetaLM**: tratar LLMs como interface universal, adicionando módulos de percepção.
- Treinado em corpora multimodais da web: texto puro, pares imagem–legenda e dados intercalados.
- Suporta linguagem, percepção–linguagem e tarefas de visão em zero-shot e few-shot.
- Permite raciocínio multimodal, diálogos visuais e interações em múltiplas modalidades.

2 Kosmos-1: A Multimodal Large Language Model

Input Representation

- Sequência linearizada com tokens especiais <image> e </image> para delimitar embeddings de imagens.
- Texto e imagens tratados de forma unificada como tokens no Transformer.
- Encoder de visão pré-treinado (CLIP ViT-L/14) com módulo Resampler para reduzir embeddings.

Architecture

- Backbone é um Transformer causal (decodificador) com ~1,6B parâmetros.
- Variantes:
  - **MAGNETO Transformer** para maior estabilidade de treino.
  - **XPOS relative position encoding** para lidar com contextos longos.
- Treinado para **next-token prediction** em dados multimodais.

Training Objective

- Previsão do próximo token, incluindo tokens textuais e alinhamento de modalidades.
- Multimodal LM training favorece capacidades emergentes úteis para downstream.

3 Model Training

Multimodal Training Data

- **Text corpora**: The Pile (subconjuntos), Common Crawl (CC-2020-50, CC-2021-04), Wikipedia, RealNews, Books, Gutenberg, CC-Stories etc. (~360B tokens).
- **Image–caption pairs**: LAION-2B (inglês), LAION-400M, COYO-700M, Conceptual Captions (CC3M, CC12M).
- **Interleaved image–text data**: 71M páginas web filtradas com texto+imagens do Common Crawl.

Training Setup

- Modelo com 24 camadas, hidden size 2048, 32 heads.
- Treinado com 300k steps (~360B tokens).
- AdamW, LR inicial 2e-4 com warmup e decaimento linear.
- Resolução de imagem: 224×224.

Language-Only Instruction Tuning

- Adiciona ajuste fino em dados textuais de instrução (Unnatural Instructions, FLANv2).
- Melhora a capacidade de seguir instruções também em tarefas multimodais.

4 Evaluation

Perception-Language Tasks

- Avaliado em captioning (COCO, Flickr30k) e VQA (VQAv2, VizWiz).
- Supera Flamingo em zero-shot captioning (mesmo com modelo menor).
- Em few-shot VQA, é competitivo com Flamingo, mostrando robustez em datasets complexos como VizWiz.

IQ Test: Nonverbal Reasoning

- Avaliado em Raven's Progressive Matrices (50 exemplos).
- Primeiro modelo a resolver esse teste em **zero-shot**, mostrando raciocínio não verbal emergente.

OCR-Free Language Understanding

- Testado em Rendered SST-2 (sentiment analysis em imagens de texto) e Hateful Memes.
- Supera CLIP e Flamingo sem usar OCR explícito → mostra habilidade embutida de "ler" imagens.

Web Page Question Answering

- Avaliado no WebSRC, que exige compreender estrutura de páginas web (layout, tabelas, listas).
- Kosmos-1 supera baseline textual, indicando que percepção visual melhora compreensão estrutural.

Multimodal Chain-of-Thought Prompting

- Extende o CoT prompting para entradas multimodais.
- Geração de racional intermediário ajuda em tasks como Rendered SST-2, melhorando precisão.

Zero-Shot Image Classification

- Avaliado no ImageNet (1k classes).
- Kosmos-1 supera GIT e CLIP em zero-shot top-1 accuracy.
- Chain-of-thought multimodal melhora ainda mais.

Zero-Shot Image Classification with Descriptions

- Com descrições verbais adicionais (ex: pássaros semelhantes), acurácia sobe de ~61% para ~90%.
- Mostra que descrições contextuais ajudam alinhamento visão–linguagem.

Language Tasks

- Avaliado em StoryCloze, HellaSwag, Winograd, Winogrande, PIQA, BoolQ, CB, COPA.
- Desempenho comparável a LLM treinado só em texto, e melhor em few-shot.
- Mostra que manter texto no treino não prejudica tarefas de NLP puras.

Cross-Modal Transfer

- **De linguagem para multimodal:** instruction tuning em texto melhora captioning e VQA.

- **De multimodal para linguagem:** Kosmos-1 supera LLMs em raciocínio de conhecimento visual (cor, tamanho de objetos).

5 Conclusion

- Kosmos-1 é um passo de LLMs para MLLMs, capaz de perceber modalidades, seguir instruções e aprender em contexto.
- Suporta uma gama ampla de tarefas: linguagem, percepção–linguagem, visão e raciocínio não verbal.
- Mostra emergências como OCR implícito, commonsense visual e CoT multimodal.
- Futuro: escalar o modelo, integrar fala, e usá-lo como interface unificada multimodal (incluindo geração de imagens condicionada por instruções).

Tabela de Datasets

Datasets utilizados no treinamento dos modelos estudados.

Nome do Dataset	Paper(s)	Tipo de Dados	Quantidade de Dados	Uso (Pré-treino ou Fine-tuning)
ImageNet (ILSVRC 2012)	DeViSE, CLIP, VirTex, SigLIP2	Imagens + rótulos de classe	1.2M imagens, 1k classes	Fine-tuning/avaliação em todos; pré-treino em DeViSE e VirTex
COCO Captions	VirTex, Flamingo, Kosmos-1	Imagens + legendas densas	118k imagens × 5 legendas	Pré-treino (VirTex); Fine-tuning/avaliação (Flamingo, Kosmos-1)

Flickr30k	CLIP, Kosmos-1, Flamingo	Imagens + legendas	31k imagens, 5 legendas cada	Fine-tuning/avaliação
LAION-400M	CLIP, ALIGN, Kosmos-1, SigLIP1, SigLIP2	Pares imagem-texto (web)	400M pares	Pré-treino
LAION-2B (EN subset)	Kosmos-1, SigLIP2	Pares imagem-texto	2B pares	Pré-treino
COYO-700M	Kosmos-1	Pares imagem-texto	700M pares	Pré-treino
Conceptual Captions (CC3M / CC12M)	ALIGN, Kosmos-1	Imagem + legenda web	3M / 12M pares	Pré-treino
The Pile	Kosmos-1	Texto	825 GB (22 corpora)	Pré-treino (texto puro)
Common Crawl (CC), RealNews, CC-Stories	Kosmos-1	Texto web filtrado	Bilhões de tokens	Pré-treino
YFCC-100M	DeViSE, VirTex, CLIP	Flickr imagens + tags	100M imagens	Pré-treino alternativo (citados, nem sempre usados diretamente)
WebLI	SigLIP1, SigLIP2	Pares imagem-texto, 109 línguas	10B imagens + 12B textos	Pré-treino principal

LiT Dataset	SigLIP1	Pares imagem-texto (Google)	~400M pares	Pré-treino
Visual Genome	Flamingo	Imagens + regiões + QA	108k imagens	Fine-tuning multimodal
OKVQA / VQAv2 / VizWiz	Kosmos-1, Flamingo	QA sobre imagens	80k / 200k / 31k QAs	Fine-tuning/avaliação
Hateful Memes	Kosmos-1, Flamingo	Memes multimodais	10k pares	Avaliação
BookCorpus, Wikipedia	DeViSE	Texto	1B+ tokens	Pré-treino embeddings de texto
JFT-300M (privado Google)	ALIGN	Imagens + rótulos automáticos	300M imagens	Pré-treino (privado)
IG-3.5B (Instagram hashtags)	VirTex (citado)	Imagens + hashtags	3.5B	Pré-treino alternativo (weak supervision)
WebPage QA (WebSRC)	Kosmos-1	Perguntas + HTML	0.2M pares	Avaliação
Raven's Progressive Matrices (Synthetic)	Kosmos-1	Imagens abstratas + questões	–	Benchmark não supervisionado

MSCOCO Detection / Segmentation	VirTex	Imagens + bounding boxes/máscaras	118k	Fine-tuning

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 24 de set. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

- Leitura dos papers para finalizar a "Linha do tempo" de trabalhos que construí:
  - FLAVA: A Foundational Language And Vision Alignment Model
    - 2022
    - Alinhamento visão-linguagem que combina pré-treino unimodal e multimodal com objetivos contrastivos, mascaramento e matching.
  - SIMVLM: SIMPLE VISUAL LANGUAGE MODEL PRE- TRAINING WITH WEAK SUPERVISION
    - 2022
    - Framework de pré-treino visão-linguagem
    - Modelo encoder–decoder baseado em Transformer
  - Visual Instruction Tuning - Llava:
    - 2023
    - Modelo multimodal que conecta CLIP ao Vicuna, treinado com instruções geradas por GPT-4. Alcança capacidades emergentes de chat visual e raciocínio.
  - Todos os resumos (em tópicos e divididos conforme divisão do paper) estão na pasta: Resumos - papers

- Leitura do survey: Multimodal Large Language Models for Text-rich Image

Understanding: A Comprehensive Review - Findings ACL 2025.

- Resumo em tópicos:
  Multimodal Large Language Models for Text-rich Image Underst…
- O survey apresenta uma revisão sistemática dos Multimodal Large Language Models (MLLMs) para compreensão de Text-rich images, cobrindo arquiteturas, pipelines de treinamento, datasets e benchmarks, além de discutir desafios e tendências futuras da área.
- Text-rich Image Understanding (TIU) combina percepção (detecção, OCR, layout, fórmulas) e entendimento (raciocínio semântico).
- Traz imagens relevantes que ajudam na compreensão geral.

- Percebi que ler esses papers da “Linha do tempo” realmente contribuiu bastante para minha percepção e entendimento geral. Trabalhos como CLIP são usados/adaptados em trabalhos recentes.

- Durante a leitura do survey e pesquisas que realizei com o Google Scholar, separei alguns papers (tanto surveys como aplicações que achei interessante).
  - Reuni todos eles em um arquivo para, posteriormente, realizar uma curadoria. Trabalhos para análise futura

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Leitura: Unified Multimodal Understanding and Generation Models: Advances, Challenges, and Opportunities - Survey (33 páginas, 2025).
- Leitura: MME-Survey: A Comprehensive Survey on Evaluation of Multimodal LLMs - Survey (25 páginas, 2024)
- Curadoria de trabalhos relevantes.

Observação: **[caso precise fazer alguma observação, de qualquer “natureza”]**

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Instituto de Informática (INF) - UFG
Alameda Palmeiras, Quadra D, Câmpus Samambaia
Fone: (62) 3521-1181 / Fax: (62) 3521-1182

Anotações sobre o Paper FLAVA: A Foundational Language And Vision Alignment Model

Abstract

- Modelos de visão e visão-linguagem atuais dependem de pré-treino visiolinguístico em larga escala.
- Normalmente são apenas contrastivos (cross-modal) ou multimodais (early fusion), mas não ambos.
- Propõe-se um modelo universal que funcione bem em visão, linguagem e multimodalidade.
- FLAVA é apresentado como esse modelo fundacional, alcançando bons resultados em 35 tarefas distintas.

1. Introduction

- Pré-treinos em larga escala trouxeram avanços para visão e visão-linguagem.
- Modelos como CLIP e ALIGN mostraram que supervisão em linguagem natural gera representações visuais de alta qualidade.
- Porém, modelos contrastivos têm limitações para problemas multimodais.
- Modelos multimodais com early fusion geralmente sacrificam desempenho unimodal.
- Um modelo fundacional ideal deve ser bom em visão, linguagem e em raciocínio multimodal.
- FLAVA combina informações de múltiplas modalidades em uma arquitetura única de transformers, treinada em dados públicos e menores do que os usados em modelos equivalentes.

2. Background

- Pré-treino auto-supervisionado trouxe avanços em NLP, visão, fala e visão-linguagem.
- Modelos existentes podem ser divididos em dois grupos:
  - Dual encoders: bons para retrieval, ruins em raciocínio multimodal.
  - Fusion encoders: bons para tarefas complexas, mas fracos em visão ou linguagem puras.
- FLAVA combina as duas abordagens em um modelo único e holístico.

3. FLAVA: A Foundational Language And Vision Alignment Model

3.1 The model architecture

- Composto por três encoders baseados em ViT:
  - Image encoder: representa imagens como patches.
  - Text encoder: mesma arquitetura ViT, mas para tokens de texto.
  - Multimodal encoder: funde representações visuais e textuais com cross-attention.
- Para downstream tasks:
  - Visual recognition → saída do image encoder.
  - NLP → saída do text encoder.
  - Multimodal reasoning → saída do multimodal encoder.

3.2 Multimodal pretraining objectives

- Global contrastive (LGC): similar ao CLIP, mas com backprop global entre GPUs.
- Masked multimodal modeling (MMM): máscara tanto em patches de imagem quanto em tokens de texto, prevendo o código visual ou token mascarado.
- Image-text matching (ITM): classifica se par imagem-texto corresponde.

3.3 Unimodal pretraining objectives

- Masked image modeling (MIM): máscara em patches de imagem e reconstrução com dVAE.
- Masked language modeling (MLM): máscara em 15% dos tokens de texto, reconstrução via encoder de texto.
- Encoders podem ser inicializados de pré-treinos unimodais (DINO para visão, MLM em CCNews/BookCorpus para texto).
- Estratégia conjunta: pré-treinar unimodalmente e depois treinar em dados uni e multimodais juntos.

3.4 Implementation details

- Uso de batch grande (8192), alta weight decay, warmup longo (10k steps).
- AdamW como otimizador.
- Arquitetura baseada em ViT pré-norm, mais robusta que BERT clássico.
- Implementado em MMF e fairseq, usando FSDP e FP16.

3.5 Data: Public Multimodal Datasets (PMD)

- Conjunto de 70M pares imagem-texto de datasets públicos: COCO, SBU, Localized Narratives, Conceptual Captions, VG, WIT, CC12M, RedCaps, YFCC.
- Dados abertos, garantindo reprodutibilidade.

4. Experiments

- Avaliação em 35 tarefas: 22 de visão, 8 de NLP (GLUE) e múltiplas multimodais (VQA, SNLI-VE, Hateful Memes, Flickr30K, COCO).
- Ablations mostram:
    - Contrastive apenas (tipo CLIP) é forte, mas inferior ao FLAVA completo.
    - Adição de MMM e ITM melhora multimodal e NLP.
    - Dados unimodais extras (ImageNet, CCNews, BookCorpus) ajudam NLP.

    - Inicialização unimodal com DINO e MLM dá ganhos em todas as modalidades.
- Resultados:
    - FLAVA supera modelos treinados em dados públicos (ViLT, ALBEF, UNITER, etc.).
    - Compete com BERT em GLUE, mesmo não sendo especializado.
    - Em multimodal, supera CLIP treinado em mesmo dataset (PMD).
    - Usa 70M pares, contra 400M do CLIP e 1.8B do SimVLM, mas ainda assim tem desempenho competitivo.

4.1 Comparison to state-of-the-art models

- Supera métodos multimodais com dados públicos em linguagem e multimodalidade.
- Fica levemente abaixo de CLIP em tarefas puramente visuais, mas melhor em NLP e multimodal.
- Mostra que combinar objetivos unimodais e multimodais resulta em representações mais gerais e transferíveis.

5. Conclusion

- FLAVA é um modelo fundacional para visão e linguagem, treinado em dados públicos.
- Bom desempenho em visão, linguagem e multimodalidade simultaneamente.
- Introduz objetivos novos e combina pré-treinos unimodais e multimodais.
- Aponta para modelos generalistas, abertos e reprodutíveis.
- Limitações: vieses herdados dos datasets, ausência de dados com texto em cena (OCR implícito).

Anotações sobre o Paper SIMVLM: SIMPLE VISUAL LANGUAGE MODEL PRE-TRAINING WITH WEAK SUPERVISION

Abstract

- Apresenta o SimVLM, um framework minimalista de pré-treino visão-linguagem.
- Diferente de métodos anteriores, não usa anotações caras nem múltiplos objetivos auxiliares.
- Baseia-se apenas em Prefix Language Modeling (PrefixLM) aplicado em grandes quantidades de dados web com fraca supervisão.
- Treinado de ponta a ponta, alcança SOTA em múltiplos benchmarks (VQA, NLVR2, SNLI-VE, image captioning).
- Demonstra forte capacidade de generalização e transferência zero-shot, incluindo VQA aberto e transferência entre modalidades.

1. Introduction

- Avanços em NLP com pré-treino auto-supervisionado inspiraram a busca por contrapartes multimodais.
- Modelos existentes de VLP costumam depender de detectores de objetos e perdas específicas de tarefa, tornando o treino complexo e pouco escalável.
- Outra linha de pesquisa usa dados web ruidosos, mas foca em tarefas específicas (ex.: CLIP em classificação/retrieval).
- O objetivo é criar um modelo simples, escalável, com capacidade zero-shot e que funcione bem no paradigma de pré-treino + fine-tuning.
- Proposta: SimVLM, que usa apenas PrefixLM com dados ruidosos de pares imagem-texto e texto puro, evitando módulos extras.

2. Related Work

- Grande parte dos VLPs anteriores depende de ROI features de Faster R-CNN treinados em Visual Genome, aumentando custo e reduzindo escalabilidade.
- Métodos sem detecção existem, mas geralmente usam datasets pequenos e limpos, com pouco zero-shot.
- Objetivos anteriores incluem ITM, masked region classification, contrastive loss e captioning supervisionado.
- Isso resulta em múltiplas perdas combinadas, complicando a otimização.
- SimVLM adota abordagem minimalista: apenas PrefixLM em imagens cruas + textos da web.

3. SimVLM

3.1 Background

- Pré-treino textual comum: Masked LM (BERT), ou LM autoregressivo (GPT).
- MLM dá representações contextuais, mas carece de capacidade generativa.
- LM autoregressivo facilita generalização zero-shot e geração aberta.
- VLPs anteriores usaram MLM, mas LM generativo foi pouco explorado.

3.2 Proposed Objective: Prefix Language Modeling

- PrefixLM permite atenção bidirecional no prefixo e fatoração autoregressiva no sufixo.
- Intuitivamente, a imagem funciona como prefixo para o texto.
- Para cada par imagem-texto, a sequência visual é usada como prefixo, seguido do texto alvo.
- Combina benefícios de MLM (contextualização bidirecional) e LM (geração).

3.3 Architecture

- Backbone Transformer encoder-decoder.
- Imagem processada em patches com convolução inicial (ResNet) para contextualização.
- Texto tokenizado via SentencePiece.
- Embeddings posicionais separados para imagem e texto, mais atenção relativa 2D para patches.
- Sem embeddings de tipo de modalidade (não trazem ganho).

3.4 Datasets

- Pré-treino em ALIGN (1.8B pares imagem-texto da web, sem filtragem pesada).
- Uso adicional de C4 (800GB de texto web) para complementar o ruído do texto em alt-text.
- Formulação unificada do PrefixLM reduz discrepância entre modalidades.

4. Experiments

4.1 Setup

- Três variantes: Base, Large, Huge, seguindo tamanhos de ViT.
- Pré-treino ~1M steps com batches de 4096 pares imagem-texto + 512 docs de texto puro, em 512 TPUs v3.
- Fine-tuning em 6 benchmarks multimodais: VQA, SNLI-VE, NLVR2, CoCo, NoCaps, Multi30k.

4.2 Comparison with Existing Approaches

- SimVLM supera todos os VLPs anteriores (LXMERT, UNITER, OSCAR, VinVL etc.).
- Base já bate modelos grandes; Huge melhora ~4 pontos em relação ao SOTA no VQA.

- Também mostra avanços em captioning e tradução multimodal, mesmo sem otimização específica (CIDEr).

4.3 Zero-Shot Generalization

Zero-shot/Few-shot Image Captioning

- Sem fine-tuning, SimVLM gera captions detalhados em COCO e NoCaps.
- No NoCaps, supera modelos supervisionados em out-of-domain.
- Captura conceitos finos, como marcas de carros, e descreve cenas complexas.
- Few-shot (1% dos dados) já se aproxima de modelos supervisionados.

Zero-shot Cross-Modality Transfer

- Modelo fine-tunado em texto puro pode transferir para tarefas VL (ex.: NLI → SNLI-VE).
- Também transfere entre línguas e modalidades (tradução EN-DE no Multi30k com imagens).
- Mostra que scaling de dados ruidosos permite emergir cross-modality transfer.

Open-ended VQA

- VQA tradicional usa classificação fechada com 3.129 respostas.
- SimVLM pode gerar respostas abertas, generalizando melhor para casos out-of-domain.
- Supera baselines discriminativos e generativos, especialmente em respostas raras.
- Pré-treino adicional no WIT melhora capacidade de VQA aberto zero-shot.

4.4 Analysis

- Encoder-decoder supera decoder-only.

- PrefixLM melhor que LM simples ou span corruption.
- Dados de texto puro compensam ruído de alt-text e aumentam qualidade.
- Data scaling crítico: 10% ALIGN reduz desempenho.
- Convolução inicial (3 blocos ResNet) é essencial para boa performance multimodal.

5. Conclusion

- SimVLM é um framework simples, sem detecção de regiões nem múltiplos objetivos.
- Treinado de ponta a ponta apenas com PrefixLM, atinge SOTA em VL.
- Mostra forte capacidade de generalização zero-shot e cross-modality.
- Aponta caminho para pré-treino generativo como alternativa promissora ao MLM.

Anotações sobre o Paper LLaVA - Vision Instruction Tuning

Visual Instruction Tuning (LLaVA) - 2023

Abstract

- O trabalho apresenta o LLaVA, o primeiro modelo multimodal de instruções visuais.
- Usa GPT-4 apenas em texto para gerar dados de instruções multimodais (imagem + linguagem).
- LLaVA conecta CLIP (encoder visual) ao Vicuna (LLM) em um modelo end-to-end treinado em dados sintéticos.
- Introduz benchmarks específicos (LLaVA-Bench) para avaliação de instruções visuais.

- Mostra desempenho impressionante, com 85,1% da pontuação relativa ao GPT-4 em benchmark sintético e 92,53% em ScienceQA (SoTA).

Introduction

- A motivação é criar um assistente multimodal de uso geral, capaz de seguir instruções em visão e linguagem.
- Modelos de visão atuais são poderosos (classificação, detecção, segmentação, geração), mas limitados em interatividade e adaptabilidade a instruções explícitas.
- LLMs funcionam como interfaces universais, capazes de receber instruções em linguagem natural e generalizar para múltiplas tarefas.
- O trabalho propõe visual instruction tuning, estendendo a técnica de *instruction tuning* da NLP para o espaço multimodal.
- Principais contribuições:
    - Geração de dados de instrução multimodal usando GPT-4.
    - LLaVA, modelo multimodal baseado em CLIP + Vicuna.
    - Dois benchmarks novos para avaliação de instruções visuais.
    - Liberação aberta de código, dados e checkpoints.

Related Work

- Dividido em duas áreas principais:
    - Agentes multimodais de instruções: modelos end-to-end para navegação visual, edição de imagens e sistemas coordenados via LLMs (Visual ChatGPT, MM-REACT, ViperGPT).
    - Instruction tuning em NLP: InstructGPT, FLAN-T5, FLAN-PaLM, OPT-IML; mostram melhora em generalização zero-shot e few-shot.
- Em multimodalidade, Flamingo, BLIP-2, FROMAGe, KOSMOS-1 e PaLM-E são precursores, mas não foram afinados explicitamente com instruções visuais.

- Diferença entre *visual instruction tuning* (melhorar habilidade de seguir instruções) e *visual prompt tuning*(eficiência paramétrica).

GPT-assisted Visual Instruction Data Generation

- O desafio é a escassez de dados de instruções multimodais.
- Proposta: converter pares imagem-texto em formato de instrução usando GPT-4.
- Representações simbólicas de imagens:
  - Captions (descrições textuais).
  - Bounding boxes (objetos e posições).
- Três tipos de dados gerados:
  - Conversação (perguntas sobre conteúdo da imagem).
  - Descrição detalhada.
  - Raciocínio complexo (questões que exigem inferência lógica).
- Dataset final: 158k amostras (58k conversação, 23k descrição detalhada, 77k raciocínio complexo).
- GPT-4 produziu instruções de maior qualidade que ChatGPT, principalmente em raciocínio espacial.

Visual Instruction Tuning

Architecture

- Baseado no CLIP ViT-L/14 como encoder visual.
- Projeção linear transforma embeddings visuais em tokens compatíveis com embeddings do Vicuna.
- Arquitetura leve e simples, favorecendo iteração rápida em experimentos.
- Possível extensão futura para mecanismos mais sofisticados (cross-attention, Q-former).

Training

- Stage 1: Pre-training para alinhamento de features

    - Dataset: CC3M filtrado (595k pares).
    - Objetivo: alinhar embeddings visuais ao espaço do LLM (treina apenas a projeção linear).
- Stage 2: Fine-tuning end-to-end
    - Congela encoder visual, treina projeção + Vicuna.
    - Dois cenários:
        - Chat multimodal: afinado em 158k instruções.
        - ScienceQA: afinado em dados multimodais de raciocínio científico (21k questões).

Experiments

Multimodal Chatbot

- LLaVA comparado a GPT-4, BLIP-2 e OpenFlamingo em tarefas de compreensão visual.
- Resultados:
    - LLaVA segue instruções de forma mais precisa que BLIP-2 e OpenFlamingo.
    - Em benchmarks, alcança 85,1% da performance relativa ao GPT-4.
    - Desempenho robusto em imagens fora do domínio, mostrando generalização.
- Novos benchmarks:
    - LLaVA-Bench (COCO): 90 perguntas de 30 imagens, avaliado com GPT-4 como juiz.
    - LLaVA-Bench (In-the-Wild): 24 imagens complexas com 60 perguntas.
- LLaVA supera BLIP-2 e OpenFlamingo em até 48 pontos percentuais.

ScienceQA

- Benchmark com 21k questões multimodais (ciências naturais, sociais, linguagem).

- LLaVA alcança 90,92% de acurácia, próximo ao SoTA MM-CoT (91,68%).
- Combinação com GPT-4 melhora ainda mais para 92,53% (novo SoTA).
- Ablations:
    - Usar feature antes da última camada do CLIP é melhor (+0,96%).
    - Pre-training é crítico (+5% em relação a treinar do zero).
    - Modelo 13B supera 7B (+1%).
    - Estratégia *reasoning-first* acelera convergência, mas não melhora resultado final.

Conclusion

- O trabalho mostra a viabilidade de visual instruction tuning.
- LLaVA demonstra capacidades emergentes de chat multimodal e raciocínio visual.
- Estabelece novo SoTA em ScienceQA e benchmarks próprios.
- Aponta caminho para agentes multimodais mais capazes, unindo alinhamento de instruções com visão e linguagem.

Anotações sobre o Paper BLIP - Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation

BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation

Abstract

- O trabalho apresenta o BLIP, um novo framework de pré-treino visão-linguagem (VLP) unificado.
- Diferente de modelos anteriores, o BLIP é capaz de lidar tanto com tarefas de entendimento (retrieval, VQA) quanto de geração (captioning).
- Propõe duas inovações:

    - Multimodal Mixture of Encoder-Decoder (MED), arquitetura que atua como encoder, text encoder condicionado em imagem ou text decoder condicionado em imagem.
    - CapFilt, método de bootstrapping de dados que usa um captioner para gerar descrições sintéticas e um filtro para remover textos ruidosos.
- BLIP alcança SOTA em diversas tarefas (retrieval, captioning, VQA, visual reasoning, dialog) e mostra forte generalização em tarefas vídeo-linguagem no zero-shot.

Introduction

- Pré-treino multimodal avançou significativamente, mas os métodos existentes tinham limitações.
- Modelos encoder-only (CLIP, ALBEF) são bons em entendimento mas não transferem bem para geração.
- Modelos encoder-decoder (SimVLM) conseguem gerar texto, mas não são competitivos em retrieval.
- Além disso, dados da web trazem muito ruído: alt-texts não descrevem fielmente as imagens.
- BLIP busca resolver ambos problemas com uma arquitetura flexível (MED) e um pipeline de dados melhorado (CapFilt).

Related Work

- Vision-Language Pre-training: base em pares imagem-texto da web, mas com forte presença de ruído.
- Knowledge Distillation: CapFilt pode ser visto como distilação, onde o captioner fornece conhecimento semântico e o filtro remove ruído.
- Data Augmentation: uso de modelos generativos para sintetizar dados, aqui aplicado para legendas multimodais em larga escala.

Method

Model Architecture (MED)

- Base no Vision Transformer (ViT) como image encoder.
- Text encoder baseado no BERT.
- Três modos de operação:
  - Unimodal encoder: usa contraste imagem-texto (ITC).
  - Image-grounded text encoder: aplica cross-attention e é treinado com Image-Text Matching (ITM).
  - Image-grounded text decoder: usa causal self-attention e é treinado com Language Modeling (LM).
- Encoder e decoder compartilham parâmetros, exceto nas camadas de self-attention.

Pre-training Objectives

- ITC: aproxima embeddings de pares corretos de imagem-texto e distancia negativos.
- ITM: classifica se uma imagem e um texto são correspondentes.
- LM: gera descrições textuais condicionadas em imagens.

CapFilt

- Captions da web são ruidosos e pouco informativos.
- O método introduz:
  - Captioner: gera legendas sintéticas de alta qualidade.
  - Filter: remove legendas ruidosas (originais e sintéticas).
- Ambos são derivados do mesmo modelo MED pré-treinado e refinados em COCO.
- O dataset final combina textos filtrados e legendas sintéticas diversificadas.

Experiments

Pre-training Setup

- Implementação em PyTorch.
- Usou ViT-B/16 e ViT-L/16.
- Pré-treino em 14M imagens (COCO, Visual Genome, Conceptual Captions, Conceptual 12M, SBU).
- Também testado com LAION (115M imagens).

CapFilt Results

- Usar apenas o captioner ou o filtro já melhora desempenho; juntos, trazem ganhos substanciais.
- Nucleus sampling para geração de captions mostrou melhor diversidade e maior ganho que beam search.
- Diversidade de captions é chave para desempenho.

Comparison with State-of-the-art

- Image-Text Retrieval: BLIP supera ALBEF (+2.7% R@1 médio no COCO). Em zero-shot no Flickr30K, também tem melhor desempenho.
- Image Captioning: supera modelos comparáveis com 14M dados e se aproxima de métodos massivos (SimVLM com 1.8B pares).
- VQA: melhora +1.6% em relação a ALBEF, competindo com SimVLM usando 13x menos dados.
- NLVR2: desempenho competitivo, próximo ao SOTA.
- Visual Dialog: alcança novo SOTA no VisDial v1.0.
- Video-Language: transferido zero-shot, BLIP supera modelos treinados especificamente em vídeo em tarefas como text-to-video retrieval.

Ablations

- Mostrar que ganhos não vêm de maior tempo de treino, mas sim da qualidade dos dados.

- Modelos precisam ser re-treinados com datasets bootstrapped, não apenas continuar do modelo antigo.
- Melhor estratégia de compartilhamento de parâmetros é compartilhar tudo, exceto self-attention.
- Evitar compartilhar parâmetros entre captioner e filter, pois gera viés e reduz qualidade da filtragem.

Conclusion

- BLIP propõe um VLP unificado capaz de lidar com entendimento e geração.
- Combinação de MED e CapFilt leva a SOTA em múltiplos benchmarks.
- Demonstra escalabilidade e generalização até para tarefas vídeo-linguagem.
- Futuros avanços incluem múltiplas rodadas de bootstrapping, múltiplas legendas por imagem e ensembles de captioners/filters.

Anotações sobre o Survey Multimodal Large Language Models for Text-rich Image Understanding: A Comprehensive Review

Multimodal Large Language Models for Text-rich Image Understanding: A Comprehensive Review

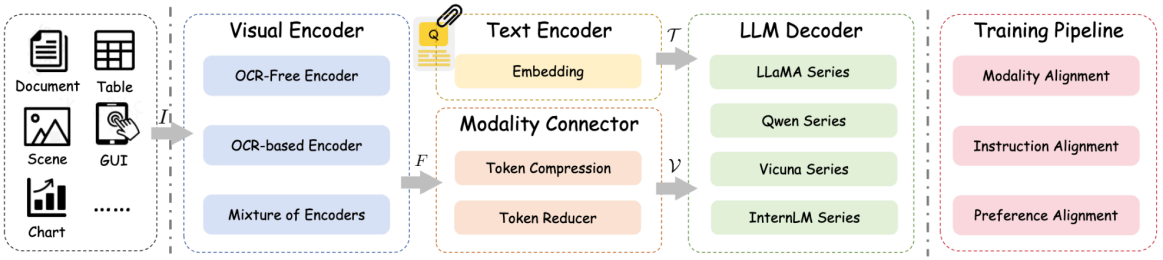


Figure 2: The general model architecture of MLLMs and the implementation choices for each component.

Abstract

- MLLMs trouxeram novas capacidades para Text-rich Image Understanding (TIU).
- Evolução dos modelos é muito rápida e difícil de acompanhar.
- Este survey organiza o campo em três frentes principais: arquiteturas e pipelines, avaliação em benchmarks e desafios e tendências futuras.

1. Introduction

- Imagens ricas em texto são cruciais no mundo real em documentos, tabelas, infográficos, formulários e cenas.
- TIU combina percepção (detecção, OCR, layout, fórmulas) e entendimento (raciocínio semântico).
- Exemplos de tarefas são DocVQA, ChartQA e TextVQA.
- Histórico: na era pré-LLM (2019–2022) os modelos eram especializados e dependentes de OCR, como LayoutLM e Donut. Na era pós-LLM (2023–) surgem MLLMs integrando encoders visuais e LLMs com maior generalização zero-shot.
- Métodos antigos tinham pouca adaptação a cenários abertos e exigiam pipelines complexos.
- Benefício dos LLMs: representação homogênea e generalização sem necessidade de fine-tuning específico.
- Surveys anteriores são fragmentados, focando em domínios isolados. Este trabalho cobre arquiteturas, pipelines, datasets/benchmarks e desafios.

2. Model Architecture

- Estrutura geral formada por três blocos: Visual Encoder, Modality Connector e LLM Decoder.
- O objetivo é extrair features visuais e textuais e alinhá-las ao espaço semântico do LLM.

Visual Encoder

- OCR-free: trabalha direto em pixels, exemplos CLIP, ConvNeXt, SAM, DINOv2, Swin-T, InternViT.
- OCR-based: injeta resultados de OCR em estratégias como input direto de texto, cross-attention ou external encoders como BLIP-2 e LayoutLMv3.
- Mixture of Encoders: combina OCR-free e OCR-based, como CLIP+SAM ou CLIP+LayoutLMv3.
- Evolução mostra aumento do uso de OCR-free e híbridos.

Modality Connector

- Responsável por alinhar embeddings de imagem e tokens textuais.
- Estratégias incluem projeções lineares ou MLP, compressão de tokens e redução de tokens.
- Técnicas específicas: cross-attention para selecionar tokens salientes, H-Reducer para convolução horizontal em textos longos, C/D-abstract unindo convolução e atenção deformável, attention pooling para eliminar tokens redundantes.

LLM Decoder

- Integra instruções do usuário com embeddings visuais alinhados.
- Modelos utilizados incluem LLaMA, Qwen, Vicuna e InternLM.
- Funções principais: raciocínio multimodal, tokenização robusta, suporte multilíngue e contexto longo.

3. Training Pipeline

- Treinamento dividido em três estágios: Modality Alignment (MA), Instruction Alignment (IA) e Preference Alignment (PA).

Modality Alignment

- Pré-treino com dados de OCR e tarefas de VQA estruturadas.
- Inclui tarefas como leitura completa de documentos, leitura parcial, predição de posição de texto.
- Estratégias para contornar limites de tokens em LLMs.
- Envolve também Text Recognition e Text Grounding.
- Parsing de elementos não textuais como gráficos (transformados em tabelas ou código), tabelas (representações em Markdown, HTML ou LaTeX) e fórmulas (LaTeX).

Instruction Alignment

- Supervised fine-tuning para alinhar instruções humanas.
- Três níveis:
    - Visual-semantic anchoring, distinguindo respostas dentro e fora da imagem.
    - Prompt diversity augmentation, gerando variações da mesma pergunta.

- Zero-shot generalization, explorando estratégias como Chain of Thought e RAG.

Preference Alignment

- Corrige divergência entre treino e inferência.
- Técnicas incluem DPO, MPO e GRPO.
- Exemplo: InternVL2.5-MPO introduziu Mixed Preference Optimization, Qwen2.5-VL aplicou DPO.

Multi-stage Training

- Pipeline típico segue MA → IA → PA.
- Preference Alignment é uma tendência recente nos modelos de ponta.

4. Datasets and Benchmarks

- O progresso em TIU dependeu de datasets e benchmarks específicos.
- Categorias principais:
  - Document: DocVQA, MP-DocVQA, FUNSD, SROIE, IIT-CDIP.
  - Chart: ChartQA, PlotQA, DVQA, ChartBench.
  - Scene: TextVQA, ST-VQA, OCR-VQA.
  - Table: TableQA, TabFact, PubTabNet, TURL.
  - GUI: ScreenQA, Screen2Words, ScreenSpot.
  - Comprehensive: OCRBench, Seed-bench, MMLongBench-Doc.

- Muitos datasets foram convertidos de tarefas tradicionais de OCR e detecção para formato VQA.
- Outros foram criados sob medida para cenários de TIU, como DocVQA, ChartQA e TextVQA.
- Benchmarks avançados avaliam capacidades específicas como long-context, cenários multilíngues ou domínios financeiros e científicos.

5. Challenges and Trends

- Ranking de melhores modelos inclui Qwen2-VL-72B, InternVL2.5 (38B, 78B, 26B) e DeepSeek-VL2.
- Observação: modelos de ponta tendem a usar OCR-free encoders, evitando redundância.
- Desafios centrais:
    - Eficiência computacional e compressão de modelos, devido ao alto custo de inferência.
    - Tokens de imagem muito longos aumentam custo e latência.
    - Entendimento de longos documentos, ainda problemático em múltiplas páginas.
    - Necessidade de benchmarks para long documents, já em desenvolvimento.
    - Multilinguality ainda limitada, com foco em inglês e línguas de alto recurso.
    - Pouca cobertura de línguas de baixo recurso, sendo necessário gerar dados sintéticos e aplicar transferência cross-lingual.

    - Natural image understanding sofre com poucos datasets realistas e desafios como ruídos, fontes artísticas e ângulos inclinados.
- Tendências promissoras incluem modelos menores com performance próxima a grandes (exemplo Mini-Monkey), compressão avançada de tokens (mPLUG-DocOwl2, TextHawk2) e treinamento multilingue com dados sintéticos.

Levantamento de Artigos de Interesse

Artigos encontrados durante levantamento biblioráfico.

Foram encontrados utilizando palavras-chave como: Modelo Multimodal, Vision, LLMs, NLP. MLLM, VLLM, Retrieval, RAG, Text-Image.

https://arxiv.org/pdf/2504.09724
https://arxiv.org/abs/2302.08641
https://arxiv.org/abs/2304.00685
https://arxiv.org/pdf/2306.13549
https://arxiv.org/abs/2410.01744
https://arxiv.org/abs/2405.14213
https://arxiv.org/abs/2410.12564
https://link.springer.com/chapter/10.1007/978-3-031-72943-0_13
https://arxiv.org/pdf/2505.02567
https://arxiv.org/abs/2408.01319
https://www.mdpi.com/2076-3417/14/12/5068
https://arxiv.org/abs/2411.15296
https://arxiv.org/abs/2405.16640
https://arxiv.org/abs/2401.13601
https://arxiv.org/abs/2408.08632
https://arxiv.org/abs/2401.10529

APÊNDICE 3

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 1 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

- Leitura: MME-RealWorld: Could Your Multimodal LLM Challenge High-Resolution Real-World Scenarios that are Difficult for Humans? - Survey (33 páginas, 2025).
  MME-RealWorld: A Multimodal Large Language Model Evaluation Benchmark in Re…
- Leitura: MME-Survey: A Comprehensive Survey on Evaluation of Multimodal LLMs - Survey (25 páginas, 2024)
  MME-Survey: A Comprehensive Survey on Evaluation of Multimodal LLMs

- Testes experimentais (apenas um input qualquer para teste, sem nenhuma finalidade específica, apenas entender o perfil deles) via API com:
  - Qwen/Qwen3-VL-Demo
  - Qwen/Qwen3-Omni-Demo
  - Intern-S1
  - Step3
  - MiniCPM-V 4.5
  - GLM-4.1V-9B-Thinking

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

- Leitura de Qwen-VL – Empowering Vision-Language Models with Multilinguality and Rich Tasks
- Leitura de NextStep-1: Toward Autoregressive Image Generation with Continuous Tokens at Scale
- Definição dos próximos passos.

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Anotações sobre o Survey MME-RealWorld: Could Your Multimodal LLM Challenge High-Resolution Real-World Scenarios that are Difficult for Humans?

MME-RealWorld: A Multimodal Large Language Model Evaluation Benchmark in Real-World Scenarios

OBJETIVO DO PAPER

Apresentar um benchmark abrangente e realista para avaliação de Modelos de Linguagem Multimodal (MLLMs), denominado MME-RealWorld, que busca medir a capacidade desses modelos em compreender e raciocinar sobre imagens de cenários do mundo real, incluindo alta resolução, detalhes complexos, relações espaciais e raciocínios multimodais contextualizados.

1. **MOTIVAÇÃO**
   Os benchmarks existentes focam em tarefas isoladas ou imagens artificiais, não refletindo a complexidade dos cenários reais.
   Há uma lacuna na avaliação de MLLMs quanto a:

- Entendimento visual detalhado e contextual.
- Raciocínio de múltiplas etapas com imagens do mundo real.
- Desempenho em ambientes naturais com múltiplos objetos e interações complexas.
  O MME-RealWorld é proposto para superar essas limitações, oferecendo um conjunto de testes realistas e diversificados.

2. **PRINCIPAIS CONTRIBUIÇÕES**

- Novo benchmark com mais de 10 mil perguntas baseadas em imagens reais e complexas.
- Cobertura de múltiplas habilidades multimodais: percepção detalhada, raciocínio visual, compreensão semântica, e coerência resposta-imagem.
- Imagens de alta resolução (até 4K), capturando riqueza visual do mundo real.
- Estrutura hierárquica de avaliação com categorias e subcategorias que refletem capacidades cognitivas multimodais.
- Avaliação padronizada de modelos abertos e fechados, fornecendo análise comparativa.

CONSTRUÇÃO DO BENCHMARK

3.1 Seleção das Imagens

- Coletadas de bancos de imagens reais com alta complexidade visual (cenas urbanas, interiores, natureza, ambientes sociais).
- Critérios: diversidade de objetos, detalhes sutis, múltiplas relações espaciais.

3.2 Tipos de Questões

As perguntas foram elaboradas para testar múltiplas dimensões:

- Percepção: identificação de objetos, cores, posições, contagem.
- Compreensão: interpretação semântica da cena.
- Raciocínio: inferências, relações, comparações, deduções.
- Multietapas: encadeamento de raciocínios, coerência global.

3.3 Processo de Anotação

- Questões criadas e revisadas manualmente por especialistas.
- Verificação de coerência entre imagem e resposta.
- Balanceamento entre tipos de questões para cobrir habilidades diversas.

3.4 Estrutura Hierárquica

O MME-RealWorld é organizado em 3 níveis:

- Nível 1: Categorias principais (Percepção, Compreensão, Raciocínio).
- Nível 2: Subcategorias (atributos, relações espaciais, cenas, lógica).
- Nível 3: Tipos específicos de tarefa (contagem, comparação, dedução).

4. AVALIAÇÃO DE MODELOS

4.1 Modelos Testados

- Fechados: GPT-4V, Gemini Pro Vision, Claude 3.
- Abertos: Qwen-VL-Plus, InternVL2, LLaVA-1.6, CogVLM, Idefics2.

4.2 Configuração Experimental

- Avaliação uniforme com prompts padronizados.
- Métrica: acurácia baseada em resposta exata e consistência com a imagem.

4.3 Resultados

- Modelos fechados superam abertos em quase todas as categorias.
- Desempenho mais fraco nas tarefas de raciocínio multietapas e percepção fina.
- Modelos abertos ainda falham em:
    - Reconhecimento de detalhes sutis.
    - Contagem precisa de objetos.
    - Compreensão de relações espaciais complexas.
- GPT-4V lidera, mas ainda com erros em cenas densas e raciocínio lógico complexo.

4.4 Análise Detalhada

- Percepção: bom desempenho em objetos grandes e claros, fraco em pequenos e sobrepostos.
- Compreensão: modelos confundem contexto semântico em cenas ambíguas.
- Raciocínio: dificuldade em inferir relações causais e temporais.

5. DESAFIOS IDENTIFICADOS
6. Alta densidade visual causa confusão e omissão de detalhes.
7. Raciocínio espacial e lógico ainda limitados.
8. Generalização para cenários não vistos permanece insuficiente.
9. Respostas inconsistentes entre execuções diferentes.
10. Dependência de prompts e dificuldade de seguir instruções complexas.
11. INSIGHTS

- A performance não escala linearmente com o tamanho do modelo.
- Modelos multimodais abertos carecem de datasets de treino realistas e variados.
- Há necessidade de treinamento supervisionado com imagens reais e instruções detalhadas.
- Benchmarks artificiais não capturam as limitações do mundo real.

Anotações sobre o Survey MME-Survey: A Comprehensive Survey on Evaluation of Multimodal LLMs

MME-Survey: A Comprehensive Survey on Evaluation of Multimodal LLMs

OBJETIVO DO PAPER

Fornecer um levantamento sistemático e detalhado sobre avaliação de Modelos de Linguagem Multimodal (MLLMs), cobrindo:

- Tipos de benchmarks e capacidades avaliadas
- Processo de construção de benchmarks

- Métodos de avaliação e métricas
- Direções futuras de pesquisa

CONTEXTO

- LLMs: grandes modelos de linguagem com capacidades emergentes (instruções, raciocínio contextual).
  MLLMs: estendem os LLMs para processar múltiplas modalidades (texto, imagem, vídeo, áudio).
  A avaliação é crítica para guiar o progresso e entender limitações.
  Diferente do paradigma tradicional de treino/teste, MLLMs exigem benchmarks abrangentes e diversificados.

ARQUITETURA E TREINAMENTO DOS MLLMs

2.1 Arquitetura típica
Encoder de modalidade (ex: visão),
Conector (alinha embeddings multimodais),
LLM backbone (gera respostas).
Modelagem autoregressiva multimodal.

2.2 Fases de treinamento

1. Pré-treinamento: alinhamento entre modalidades (ex: image-caption).
2. Instruction Tuning: aprender a seguir instruções multimodais.
3. Alignment Tuning: alinhar preferências humanas (reduzir alucinações).
4. CATEGORIAS DE BENCHMARKS
   Benchmarks dividem-se em três grandes grupos:

3.1 Capacidades Fundamentais

Avaliam percepção e raciocínio multimodal.

3.1.1 Avaliação Abrangente

Exemplos: VQA v2, VizWiz, LVLM-eHub, MME, MMBench, SEED-Bench, MM-Vet.

Foco em tarefas como VQA, raciocínio comum, contagem, localização.

Desafios: percepção detalhada, interleaving de imagem-texto, raciocínio matemático visual.

Modelos fechados (GPT-4V) ainda superam abertos, mas a lacuna está diminuindo.

3.1.2 OCR

Benchmarks: TextVQA, OCR-VQA, OCRBench, SEED-Bench-2-Plus.

Desafios: texto manuscrito, multilinguismo, texto artístico.

3.1.3 Gráficos e Documentos

Benchmarks: ChartQA, DocVQA, InfoVQA, DocGenome.

Foco em leitura estrutural, raciocínio com gráficos, longos contextos multimodais.

3.1.4 Raciocínio Matemático

Benchmarks: MathVista, MathVerse, We-Math.

Desafios: diagramas visuais complexos e decomposição de problemas.

3.1.5 Multidisciplinar

Benchmarks: ScienceQA, MMMU, CMMU, CMMMU.

Avaliam conhecimento geral e acadêmico (ciência, arte, medicina).

3.1.6 Multilíngue

Benchmarks: CMMMU (chinês), Urdu-VQA, MTVQA, M3Exam.

Desempenho melhor em idiomas latinos; fraco em línguas não ocidentais.

3.1.7 Seguir Instruções

MIA-Bench: avalia aderência a instruções complexas.

3.1.8 Multi-round QA

ConvBench, MMDU: simulação de diálogo com múltiplos turnos e imagens.

3.1.9 Multi-imagem

SparklesEval, MuirBench: raciocínio agregando várias imagens.

3.1.10 Dados Intercalados (interleaved)

VEGA, MMMU: texto e imagens misturados.

3.1.11 Alta Resolução

V*Bench, MME-RealWorld: imagens grandes e densas.

3.1.12 Grounding Visual

RefCOCO, RefCOCO+, Ref-L4: localizar objetos a partir de descrições.

3.1.13 Percepção Detalhada

FOCI, MMVP: foco em detalhes finos (cor, direção, atributos).

3.1.14 Compreensão de Vídeos

Video-MME, MVBench, LVBench: contexto temporal, vídeos longos.

Limitações: percepção temporal, raciocínio sequencial.

3.2 Autoanálise do Modelo

Avalia comportamentos e limitações internas.

3.2.1 Alucinações

Benchmarks: POPE, GAVIE, HallusionBench, AMBER, VideoHallucer.

Causas: deficiência visual e vieses de linguagem.

Alucinações comuns: objetos inexistentes, atributos errados.

3.2.2 Viés

VLBiasBench, Bingo: viés social, regional e espúrio.

Modelos abertos têm mais viés que proprietários.

3.2.3 Segurança

AttackVLM, MultiTrust, VLLM-safety-bench: robustez adversarial, jailbreaks.

Modelos vulneráveis a prompts maliciosos visuais.

3.2.4 Raciocínio Causal

CELLO: avalia entendimento de causa e efeito; desempenho fraco.

3.3 Aplicações Estendidas

Avaliação em domínios específicos.

3.3.1 Medicina

Benchmarks: VQA-RAD, PMC-VQA, OmniMedVQA.

Desempenho baixo; falta de dados médicos de qualidade.

3.3.2 Emoção

EmoBench, FABA-Bench: reconhecimento emocional multimodal.

Fine-tuning especializado melhora resultados.

3.3.3 Sensoriamento Remoto

RSVQA, RSIEval, VRSBench: interpretação de imagens de satélite.

MLLMs ainda fracos; fine-tuning ajuda.

3.3.4 Agentes

AppAgent, MobileEval, GPT4Tools: execução de tarefas com apps.

Limitações: planejamento, execução precisa.

3.3.5 Geração de Código

ChartMimic, Web2Code: conversão de gráficos/web em código.

Modelos abertos ainda inferiores aos fechados.

3.3.6 Interface Gráfica (GUI)

ScreenQA, Widget Captioning, Screen2Words: raciocínio em UIs.

Dificuldade em ícones pequenos e layouts complexos.

3.3.7 Transferência

VLAA, BenchLMM: generalização em estilos novos.

Robustez não correlaciona diretamente com tamanho do modelo.

3.3.8 Edição de Conhecimento

MMEdit, VLKEB: atualização de fatos multimodais sem retreino.

3.3.9 IA Incorporada (Embodied AI)

EQA, Ego4D, EMQA: agentes em ambientes 3D.

Limitações: percepção espacial, planejamento preciso.

3.3.10 Direção Autônoma

BDD-X, Talk2Car, DRAMA: compreensão de cenas de trânsito.

Desempenho limitado em raciocínio espacial e segurança.

- MÉTODOS DE AVALIAÇÃO
  Baseados em humanos: anotadores humanos avaliam respostas.
  Baseados em LLMs/MLLMs: avaliadores automáticos.
  Baseados em scripts: métricas objetivas (acurácia, BLEU, F1).
  Toolkits: OpenCompass, VLMEvalKit, MM-Bench-Toolkit.
- MÉTRICAS
  Determinísticas: acurácia, precisão.
  - notas subjetivas por humanos/LLMs.
  Limitações: não capturam bem raciocínio intermediário.
- TENDÊNCIAS FUTURAS
  Benchmarks capability-oriented (avaliar habilidades específicas).
  Mais modalidades (áudio, 3D, tato).
  Avaliações mais explicáveis e eficientes.
  Necessidade de padronização e colaboração comunitária.

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 8 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

- Leitura do survey: Unified Multimodal Understanding and Generation Models: Advances, Challenges, and Opportunities
  - Foi uma leitura bem útil, divide Multimodal Models em Compreensão Multimodal e Geração Multimodal. Faz subdivisões interessantes, como:

    Geração Multimodal AR

    - Pixel-based Encoding - gera imagens pixel a pixel de forma autoregressiva, sem usar embeddings semânticos.
    - Semantic Encoding - gera imagens a partir de tokens semânticos aprendidos em vez de pixels brutos
    - Learnable Query Encoding - usa vetores de consulta aprendíveis para conectar texto e imagem durante a geração.
    - Hybrid Encoding - combina representações de pixels e tokens semânticos em um mesmo modelo.

- A leitura do paper me fez perceber um gap no meu aprendizado, entender melhor como os tokenizadores funcionam, já que são uma parte importante.

- Para isso, estudei o Vector Quantised Variational Autoencoder (VQVAE) e VQGAN.
  - VQGAN e VQVAE (Não é um resumo dos papers, e sim anotações gerais)
  - Por mais que não sejam exatamente modelos multimodais, são muito utilizados, então senti que era muito importante eu entender o funcionamento.

- Além disso, as classificações ajudam muito no processo, então ler sobre essas classificações e entender como e onde os trabalhos que eu já li estão é muito interessante.
- O survey também traz imagens gerais de como são as arquiteturas (como por exemplo: um encoder de texto e um de imagem e então alguma camada de projeção ou “junção”.

- Experimentação prática inicial usando o framework huggingface
  - Realizei um fine tuning do BLIP utilizando o huggingface para carregar o modelo pré-treinado: "Salesforce/blip-image-captioning-base"
  - Utilizei o dataset no huggingface "laicsiifes/coco-captions-pt-br" - o dataset coco-captions traduzido para português.
  - Utilizei o pytorch.
  - Utilizei o Google Colab com a GPU T4.

- Fiz o fine tuning com “poucos”dados (20 mil) e apenas 1 época, mas o modelo já teve uma grande diferença.
- ANTES: 'pred': 'a little girl holding a cat in her hands', 'ref': 'Uma menina segurando um gatinho ao lado de uma cerca azul.'}
- DEPOIS: 'pred': 'uma garota segurando um gato na mão.', 'ref': 'Uma menina segurando um gatinho ao lado de uma cerca azul.'}
- finetuning_blip.ipynb

- Inicialmente pensei que não havia frameworks específicos para Multimodal Models, porém, ao longo de explorações, descobri:
  - SGLang
  - vLLM

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Testar implementar um pipeline mais robusto ou replicar um experimento.
- Testar algum modelo de geração de imagens.
- Abrir e entender como funciona o vLLM e SGLang.
- Explorar e pesquisar datasets em português.
- Entender sobre o estado da arte na geração de imagens autoregressiva (papers e códigos (e frameworks)).
- Se necessário, ler mais surveys.

Observação: **[caso precise fazer alguma observação, de qualquer “natureza”]**

Eu, Daniel Machado e Bernardo nos reunimos para o Daniel nos explicar o que teve na aula de sexta, isso me ajudou bastante.

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Instituto de Informática (INF) - UFG
Alameda Palmeiras, Quadra D, Câmpus Samambaia
Fone: (62) 3521-1181 / Fax: (62) 3521-1182

VQVAE e VQGAN

- **VQVAE**

Vector Quantized Variational Autoencoder

- Comprimir imagem em códigos discretos aprendidos — em vez de contínuos como nos VAEs tradicionais.
- Pixel a pixel

- Encoder -> comprime a imagem -> transforma a imagem em vetores latentes contínuos.
- Quantização vetorial -> Codebook -> Guarda um conjunto fixo de vetores (códigos) com padrões visuais -> cada vetor latente é substituído pelo vetor mais próximo em um codebook discreto (dicionário de embeddings aprendidos).
- Decoder -> Reconstrói a imagem a partir dos vetores discretizados.

Encoder

- Imagem 128x128 -> O encoder (geralmente uma CNN) reduz o tamanho (apenas o que realmente importa) -> mapa de 32×32 com vetores de 64 números (chamados features)
- Resultado: um “mapa comprimido” da imagem.

Quantização vetorial (VQ)

- O encoder gera vetores contínuos -> não queremos infinitas possibilidades e sim categorias discretas.

  Codebook:

  - Ele é um dicionário com K vetores fixos (por exemplo, 512 vetores).
  - Cada vetor representa um “tipo de padrão visual”.
  - Para cada posição do mapa gerado pelo encoder, o modelo escolhe o vetor mais parecido do codebook. -> Quantização
- A imagem vira uma matriz de índices

Decoder

Agora o decoder recebe esses códigos (os vetores escolhidos) e tenta reconstruir a imagem original.

- Ele aprende a "remontar” a imagem usando apenas os códigos.
- Se o codebook for bom, a reconstrução sai bem parecida.

LOSS

1. Loss de reconstrução — quão parecida a imagem reconstruída está da original (usa MSE, o erro quadrático).
2. Atualização do codebook — aproxima os vetores do dicionário das saídas reais do encoder.
3. Commitment loss — força o encoder a se comprometer com um vetor, sem ficar “mudando de ideia”.

$$\mathcal{L} = \underbrace{\|x - \hat{x}\|^2}_{\text{reconstrução}} + \underbrace{\|sg[z_e(x)] - e\|^2}_{\text{atualiza codebook}} + \underbrace{\beta\|z_e(x) - sg[e]\|^2}_{\text{commitment}}$$

Depois de treinado:

- Você pode converter qualquer imagem em uma sequência de códigos (os índices do codebook);
- Esses códigos são discretos -> dá pra treinar um modelo como GPT pra prever novos códigos -> imagens.
- E esse modelo pode gerar imagens novas, token por token.

Limitações:

- As imagens reconstruídas são borradas, pois o MSE favorece médias.
- A qualidade perceptual é limitada.

- **VQ-GAN**

- O VQ-VAE aprende comparando cada pixel da imagem original com o pixel da reconstrução (erro MSE).
- E isso faz o modelo tentar “tirar a média” — o que apaga detalhes finos, texturas, brilhos, bordas, etc.
- Não é realista.

- VQ-VAE (compressão → quantização → reconstrução) + discriminator (Distingue imagens reais das reconstruídas)

- Encoder
- Quantização

- Decoder (Mas, agora, o VQ-GAN não é julgado só pelo MSE (erro de pixel))

- Discriminator
  - imagens reais (vindas do dataset)
  - quanto para imagens reconstruídas (do decoder)
  - Tenta adivinhar quais são reais e quais são falsas.

LOSS

LOSS = Lvq (igual ao do VQ-VAE) + √(lambda) Lgan

$$= \|\phi(x) - \phi(\hat{x})\|^2$$

$$= \log D(x) + \log(1 - D(\hat{x}))$$

Treinamento do Transformer

- Depois de treinar o VQ-GAN, cada imagem é representada como sequência de índices discretos do codebook.
- O Transformer é treinado de forma autoregressiva a prever o próximo índice → p(s_i | s_{<i}).

Fine-tuning do Modelo BLIP

- Uso do HuggingFace para acessar o modelo, dataset e funções de transformer e treinamento.
- Fine-tuning do Modelo BLIP em PT-BR
- dataset_name = "laicsiifes/coco-captions-pt-br"
- model_name = "Salesforce/blip-image-captioning-base"
- Durante 1 época
- TrainOutput(global_step=1563, training_loss=7.264942698805132, metrics={'train_runtime': 1848.0953, 'train_samples_per_second': 13.527, 'train_steps_per_second': 0.846, 'total_flos': 0.0, 'train_loss': 7.264942698805132, 'epoch': 1.0})
- Descrições antes do Fine -tuning:
    - [{'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000184613.jpg',
    - 'pred': 'a group of people are gathering around a field of grass',
    - 'ref': 'Uma criança segurando um guarda-chuva florido e acariciando um iaque.'},
    - {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000403013.jpg',
    - 'pred': 'a kitchen with a sink and stove in it',
    - 'ref': 'Uma cozinha estreita repleta de eletrodomésticos e utensílios de cozinha.'},
    - {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000562150.jpg',
    - 'pred': 'a little girl holding a cat in her hands',
    - 'ref': 'Uma menina segurando um gatinho ao lado de uma cerca azul.'},

- {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000360772.jpg',
- 'pred': 'a bathroom with a toilet and green walls',
- 'ref': 'Um vaso sanitário localizado em um banheiro próximo a uma pia.'},
- {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000340559.jpg',
- 'pred': 'a kitchen with two sinks and a sink',
- 'ref': 'Existem duas pias ao lado de dois espelhos.'}]

- Descrições depois do Fine-tuning:
  - [{'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000184613.jpg',
  - 'pred': 'uma mulher segurando um guarda - chuva em um campo.',
  - 'ref': 'Uma criança segurando um guarda-chuva florido e acariciando um iaque.'},
  - {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000403013.jpg',
  - 'pred': 'uma cozinha com uma cozinha e uma pia.',
  - 'ref': 'Uma cozinha estreita repleta de eletrodomésticos e utensílios de cozinha.'},
  - {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000562150.jpg',
  - 'pred': 'uma garota segurando um gato na mao.',
  - 'ref': 'Uma menina segurando um gatinho ao lado de uma cerca azul.'},

- {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000360772.jpg',
- 'pred': 'um banheiro com vaso sanitario e vaso sanitario.',
- 'ref': 'Um vaso sanitário localizado em um banheiro próximo a uma pia.'},
- {'url': 'http://images.cocodataset.org/val2014/COCO_val2014_000000340559.jpg',
- 'pred': 'uma cozinha com pias de metal e pias de lavatorio.',
- 'ref': 'Existem duas pias ao lado de dois espelhos.'}

APÊNDICE 4

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 15 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Modelos Multimodais - Imagem-Texto

- Realizei estudos sobre Difusão:
  - Vídeos e materiais: Lista - Estudos - Difusao
  - De forma geral, modelos autoregressivos tratam a imagem como uma sequência de partes individuais (por exemplo, pixels, patches ou tokens visuais) e geram cada parte condicionalmente às anteriores. Já modelos de difusão aprendem a reverter um processo de ruído progressivo, durante a inferência o modelo parte de uma imagem com ruído para uma imagem coerente (denoising). Trata a imagem como uma única parte.

- Construí um mapa/organização sobre Modelos Multimodais - Texto-Imagem:
  - Multimodal.pdf ou Link do Canva (para melhor resolução)
  - Dividi em Modelos de Compreensão de imagens, Geração e Compreensão + Geração. Coloquei o pipeline geral das arquiteturas com várias anotações específicas para cada categoria, mas também gerais. Exemplifiquei algumas técnicas utilizadas em cada etapa (envolvendo tanto autorregressivo quanto difusão).

- Reuni algumas ideias de experimentos, pesquisei os artigos e a viabilidade.

Principalmente, em cima de artigos que eu já tinha estudado. Ideias Gerais

- Após refletir sobre algumas alternativas, minha decisão foi: reproduzir o LLaVA - https://github.com/haotian-liu/LLaVA.
  - Um grande limitante da área é o custo computacional, então precisei considerar esse fator na minha escolha, apesar de não ser tão simples, o LLaVA é mais viável que muitos outros modelos como o Flamingo.
  - O LLava tem um bom resultado em benchmarks.
  - O LLava disponibiliza todo o código e dataset, tornando "possível"reproduzir.
  - Não estaria apenas na camada superficial da aplicação, e sim, na base do conhecimento. De fato, treinando e atualizando um modelo.

- Iniciei a implementação do código.

- Durante minhas pesquisas,encontrei esse trabalho que foca justamente na implementação eficiente do LLaVa:
  - LLAVA-MINI: EFFICIENT IMAGE AND VIDEO LARGE MULTIMODAL MODELS WITH ONE VISION TOKEN (ICLR 2025)
    - Contém toda a implementação em código: https://github.com/ictnlp/LLaVA-Mini

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

- Implementar, de fato, o LLaVA ou o LLaVA-Mini, conseguindo algum resultado.
- Durante minha implementação, e usando os conhecimentos adquiridos, pensar em possíveis mudanças que seja possível realizar.
- Se a implementação inicial correr bem, já pensar em outros datasets de outros domínios para teste.
- Continuar meus estudos, principalmente sobre Difusão e derivados como Flow Matching. E gerar algum material - resumo sobre o tema.

Observação: **[caso precise fazer alguma observação, de qualquer "natureza"]**

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Lista de Conteúdos Levantados sobre Modelos de Difusão

Estruturei uma lista de conteúdos para estudo de Modelos de Difusão.

Englobam vídeos, artigos e sites.

Essa semana:

- Generative Image Models (Diffusion, GAN, VAE, Flow) Explained By Google En…
  But how do Diffusion Language Models actually work?
- How I Understand Diffusion Models
- Latent Diffusion Explained Simply (with Pokémon) | by Leonardo Castorina | Towards AI

Outros materiais levantados

- Denoising Diffusion Probabilistic Models
- https://www.youtube.com/playlist?list=PL57nT7tSGAAUDnli1LhTOoCxlEPGS19vH
- The U-Net (actually) explained in 10 minutes
- Text diffusion: A new paradigm for LLMs
- The physics behind diffusion models
- Flow Matching | Explanation + PyTorch Implementation
- Diffusion Models | Paper Explanation | Math Explained
  Diffusion Models From Scratch | Score-Based Generative Models Explained | M…
- Flow Matching | Explanation + PyTorch Implementation
- Diffusion models explained in 4-difficulty levels

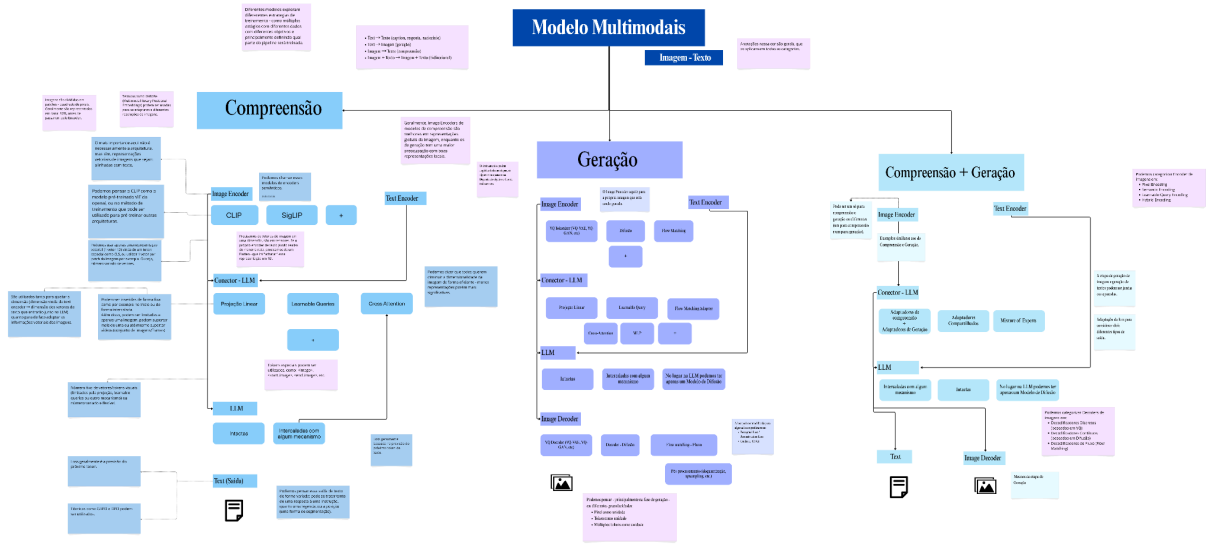
Papers:

- DALLE-2
- Stable Diffusion
- PixArt-α / PixArt-Σ
- Imagen
- FLUX
- Lumina-GPT

Mapa Mental sobre Modelos Multimodais Texto - Imagem

Mapa mental detalhado sobre Modelos Multimodais Texto-Imagem.

É subdividido em Modelos de Compreensão, Geração e Compreensão + Geração.



Primeiras Ideias de Aplicações

Registrei o brainstorming de ideias de possíveis aplicações e trabalhos interessantes.

- RAG multimodal (Ask in Any Modality: A Comprehensive Survey on Multimodal Retrieval-Augmented Generation)
- Edição de imagens - imagem como entrada + imagem como saída - como manter constância, etc. (https://arxiv.org/pdf/2501.04699)
- Técnicas de post-training - Reinforcement Learning (https://arxiv.org/pdf/2503.21758)
- Diferentes abordagens em modelos multimodais de compreensão:
  - Uso do CLIP vs SigLIP.
  - Uso do token CLS vs uso de todos os tokens dos patches como representação de imagem.
  - Diferentes métodos de adaptadores de imagem - texto (MLP, Proj. Linear, Learnable Queries, Cross-Attention)
  - Efeito dos dados - Data Mixing (muitos papers fazem essas análises) - Muito custo compuitacional - inviável. Talvez eu consiga explorar qual o MENOR conjunto de boa qualidade consigo usar.
- Métodos de evitar catastrophic forgetting em MLLMS. (Flamingo usa tan gatting)
- M-RoPE: Método de positional embeddings para imagens de diferentes tamanhos/dimensionalidades. Outras técnicas viáveis.
- Para explorar muitas dessas ideias, eu preciso primeiro conseguir implementar um modelo, opções:

- Flamingo
- BLIP-2
- Llava
- Qwen-VL

Links de trabalhos relevantes:
https://github.com/dair-iitd/MPdialog
https://github.com/Asad-Ismail/flamingo-simple
https://github.com/llava-rlhf/LLaVA-RLHF
https://www.philschmid.de/fine-tune-multimodal-llms-with-trl
https://github.com/LLaVA-VL/LLaVA-NeXT/
https://github.com/haotian-liu/LLaVA/tree/main/llava/model
https://github.com/FanqingM/MM-Eureka-V0
https://github.com/haotian-liu/LLaVA
[https://arxiv.org/abs/2410.20459
https://github.com/Knorrsche/ComicScene154
https://arxiv.org/pdf/1611.05118
https://arxiv.org/pdf/2407.03540v1
https://arxiv.org/pdf/2005.04425
https://github.com/Knorrsche/ComicScene154
https://www.cse.iitd.ac.in/~mausam/papers/acl23a.pdf]

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 23 de out. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Modelos Multimodais - Visão - Texto
Nas semanas anteriores me aprofundei nas bases teóricas, li e estudei muitos trabalhos, cosntruíndo uma linha do tempo, estudei suveys para compreender a área como um todo, estudei sobre os dados utilizados e contruí, além de resumos um "Mapa Mental"com a intenção de englobar o que aprendi.
Além disso, tive um contato prático com o fine-tuning do BLIP no Google Colab com a intenção de compreender como manipular Modelos Multimodais.

Nessa semana:

Continuei meus estudos sobre Difusão:

- Continuei me aprofundando na base teórica.
- Pesquisei artigos de geração de imagem multimodal que utilizam difusão.
- Estudos-Difusão - Materiais que embasei meus estudos e algumas anotações.

Na semana anterior, havia planejado implementar o LLaVA ou o Mini-LLaVA, porém pesquisei e constatei que era inviável com meus recursos computacionais.

Com isso, tive que iniciar novas pesquisas. Após dedicar muito tempo e esforço, reuni alguns trabalhos e um pouco da minha trajetória de ideias e pesquisa nesse documento.
Nele eu relato como realizei as pesquisas e as dificuldades que encontrei. Artigos

Após analisar dezenas de cenários e trabalhos, buscando algum que fosse viável reproduzir, o que mais se destacou foi:
TinyGPT-V: Efficient Multimodal Large Language Model via Small Backbones -> TinyGPT-V: Efficient Multimodal Large Language Model via Small Backbones

Com isso, iniciei a implementação desse trabalho, que relato o processo neste documento. TinyGPT-V

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

Continuar a implementação do TinyGPT-V, que inclui:
- Executar o treinamento
- Talvez adaptar para um domínio (usando o treinamento do “zero”ou a partir de algum checkpoint
- Talvez alterar algo na arquitetura

Observação: [caso precise fazer alguma observação, de qualquer “natureza”]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Anotações sobre Modelos de Difusão

Materiais de estudo:

- Denoising Diffusion Probabilistic Models
- Diffusion Models | Paper Explanation | Math Explained
- Diffusion models explained in 4-difficulty levels
- PixArt-Σ: Weak-to-Strong Training of Diffusion Transformer for 4K Text-to-Image Generation https://arxiv.org/pdf/2403.04692
- https://arxiv.org/pdf/2505.15809

Anotações:

Os modelos de difusão funcionam basicamente aprendendo a transformar uma imagem com ruído em uma imagem real.

- Pegar uma imagem real, adicionar ruído até ela virar puro ruído e depois treina uma rede para desfazer esse processo e reconstruir a imagem original.

O modelo aprende a reverter o ruído gradualmente, prevendo o tipo de ruído que foi adicionado em cada etapa, até chegar de volta a algo parecido com a imagem original.

Guidance -> ajuda o modelo a seguir instruções de texto com mais precisão -> por exemplo, quando o objetivo é gerar uma imagem a partir de uma descrição. Ele melhora o alinhamento entre texto e imagem.

U-Net -> trata a imagem como um mapa de features e usa atenção para misturar informações espaciais e de texto.

- Stable Diffusion utiliza U-Net

PixArt-Σ substitui U-Net por Transformers -> que processam os blocos da imagem como se fossem tokens de linguagem.

- weak-to-strong training -> o modelo começa treinando com dados e anotações mais simples (fase “weak”) e depois é refinado com dados de maior qualidade (fase “strong”).
- essa estratégia ajuda o modelo a aprender uma base sólida antes de ser exposto a dados mais complexos, o que torna o treino mais estável e eficiente

Levantamento Bibliográfico de possíveis aplicações

Pesquisei muitos e muitos papers, gastei muitas horas em busca de trabalhos que fossem reprodutíveis em uma 4090, ou no máximo em uma A100 do colab.
Grande parte dos trabalhos usavam GPUs com mais de 80GB de RAM, dezenas de A100, grandes clusters ou máquinas muito poderosas.
Além de procurar por trabalhos que fossem viáveis computacionalmente, ainda deveriam ter repositórios ou códigos open source.
Visitei inúmeros trabalhos, buscando algo que eu conseguisse ao menos iniciar a reprodução.

Usei palavras-chave (vision LLM, efficient, gpu, low…) no Google Scholar, ArxiV, Google, também usei a ferramenta de Deep Research do Chat GPT e Gemini 2.5 Pro. Além de trabalhos citados em artigos.

Algumas ideias gerais:

- RAG multimodal (busca de texto ou imagem)
- Retriever Multimodal (texto-imagem ou imagem-texto)
- VLLM
- Fine Tuning de VLLM

Apesar de muitos trabalhos tentaram usar alternativas para reduzir o custo computacional, ainda são necessários muitos recursos.

Trabalhos levantados:

- https://arxiv.org/pdf/2505.07879
- https://arxiv.org/pdf/2412.00876
- https://arxiv.org/pdf/2508.17079v1
- https://arxiv.org/pdf/2505.03181
- https://arxiv.org/pdf/2510.02270
- https://arxiv.org/pdf/2509.07488
- https://openaccess.thecvf.com/content/CVPR2024/papers/Diao_UniPT_Universal_Parallel_Tuning_for_Transfer_Learning_with_Efficient_Parameter_CVPR_2024_paper.pdf#:~:text=,modal%20tasks
- https://github.com/OpenGVLab/LLaMA-Adapter

- https://proceedings.neurips.cc/paper_files/paper/2022/file/54801e196796134a2b0ae5e8adef502f-Paper-Conference.pdf
- https://github.com/ylsung/Ladder-Side-Tuning/tree/main
- https://github.com/ylsung/VL_adapter
- https://arxiv.org/pdf/2508.15688
- https://github.com/sathiiii/microCLIP
- https://github.com/SnowNation101/Nyx
- https://aclanthology.org/2024.emnlp-main.797.pdf
- https://arxiv.org/pdf/2406.05130
- https://arxiv.org/pdf/2211.00575
- https://github.com/DavidHuji/CapDec?tab=readme-ov-file
- https://aclanthology.org/2024.emnlp-main.613.pdf
- https://arxiv.org/pdf/2309.17133
- https://arxiv.org/pdf/2412.16701v1
- https://github.com/LinWeizheDragon/Retrieval-Augmented-Visual-Question-Answering
- https://huggingface.co/learn/cookbook/multimodal_rag_using_document_retrieval_and_vlms
- https://aclanthology.org/2024.emnlp-main.62.pdf
- https://aclanthology.org/2024.emnlp-main.922.pdf
- https://arxiv.org/pdf/2210.02928
- https://arxiv.org/pdf/2312.16862v2
- https://github.com/DLYuanGod/TinyGPT-V
- https://huggingface.co/keeeeenw/MicroLlava-Qwen3-0.6B-base-siglip2-so400m
- https://github.com/luyug/GC-DPR/tree/main
- https://github.com/AssemblyAI-Community/MinImagen
- https://github.com/LinWeizheDragon/Retrieval-Augmented-Visual-Question-Answering
- https://arxiv.org/pdf/2410.10594
- https://github.com/illuin-tech/colpali/tree/main
- https://github.com/LinWeizheDragon/Retrieval-Augmented-Visual-Question-Answering
- https://proceedings.neurips.cc/paper_files/paper/2022/file/54801e196796134a2b0ae5e8adef502f-Paper-Conference.pdf
- https://arxiv.org/pdf/2304.15010
- https://arxiv.org/pdf/2509.07488
- https://github.com/yejinc00/PREMIR
- https://arxiv.org/pdf/2309.17133
- https://github.com/LinWeizheDragon/FLMR
- https://aclanthology.org/2024.emnlp-main.613.pdf
- https://github.com/kaisugi/entity-related-papers

- https://github.com/illuin-tech/colpali
- https://github.com/Osilly/dynamic_llava
- https://github.com/hiyouga/LLaMA-Factory
- https://github.com/mingllli/CLIPFit
- https://github.com/alenai97/PEFT-MLLM
- https://github.com/DavidHuji/CapDec
- https://github.com/cornstarch-org/Cornstarch
- https://anonymous.4open.science/r/MDPR-328C/README.md***
- https://github.com/castorini/UniRAG/tree/main***

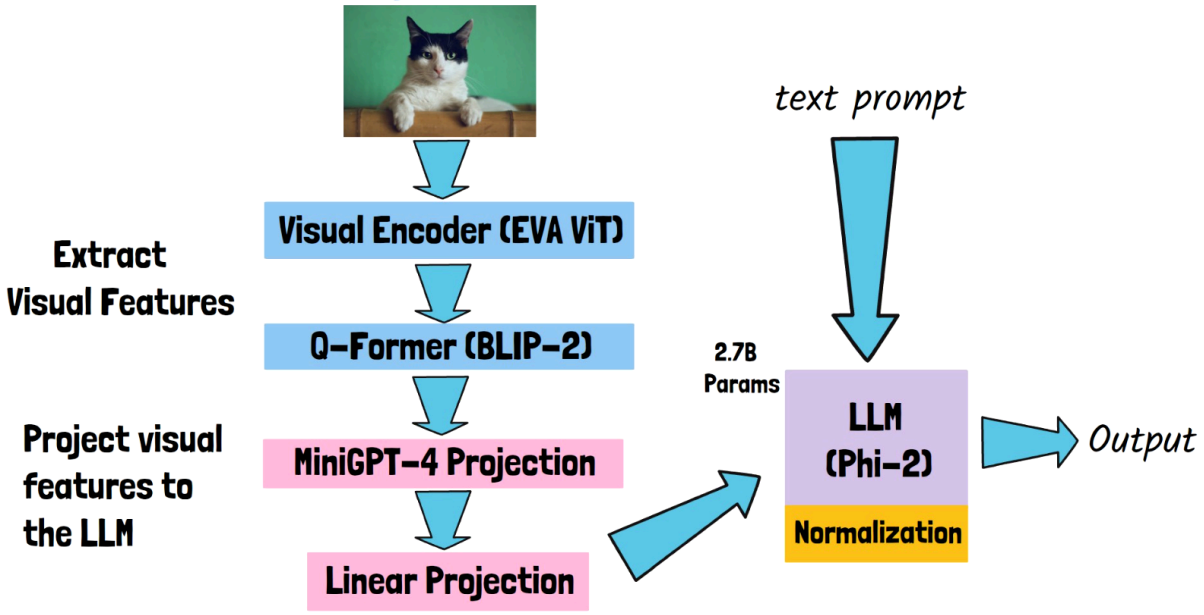
***: Maior relevância.

Experimentação Prática com Modelo TinyGPT-V

Links:

https://arxiv.org/pdf/2312.16862

https://github.com/DLYuanGod/TinyGPT-V/tree/main

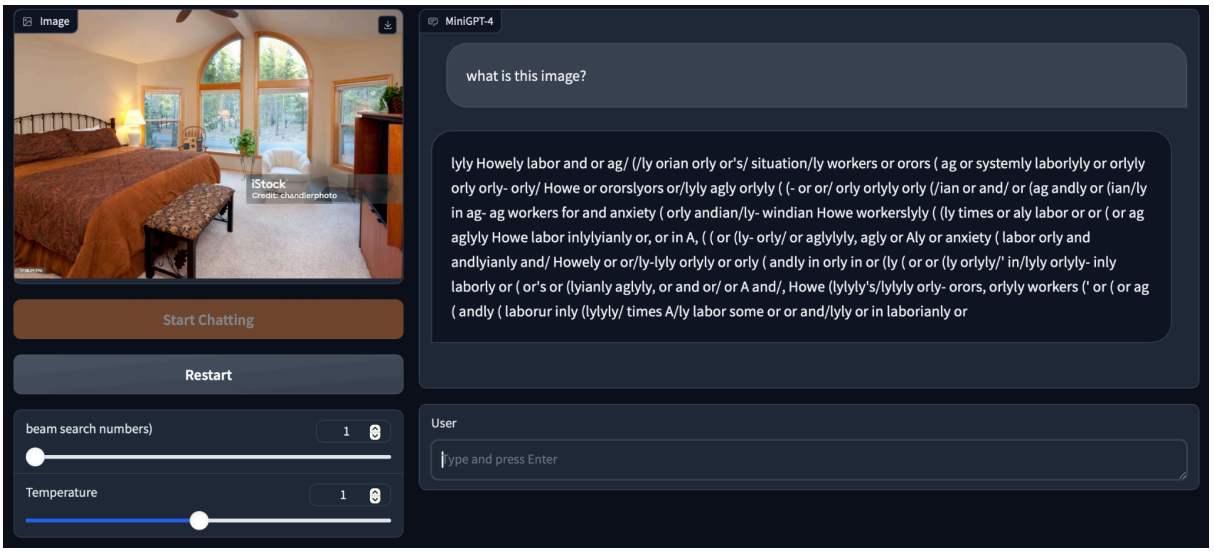


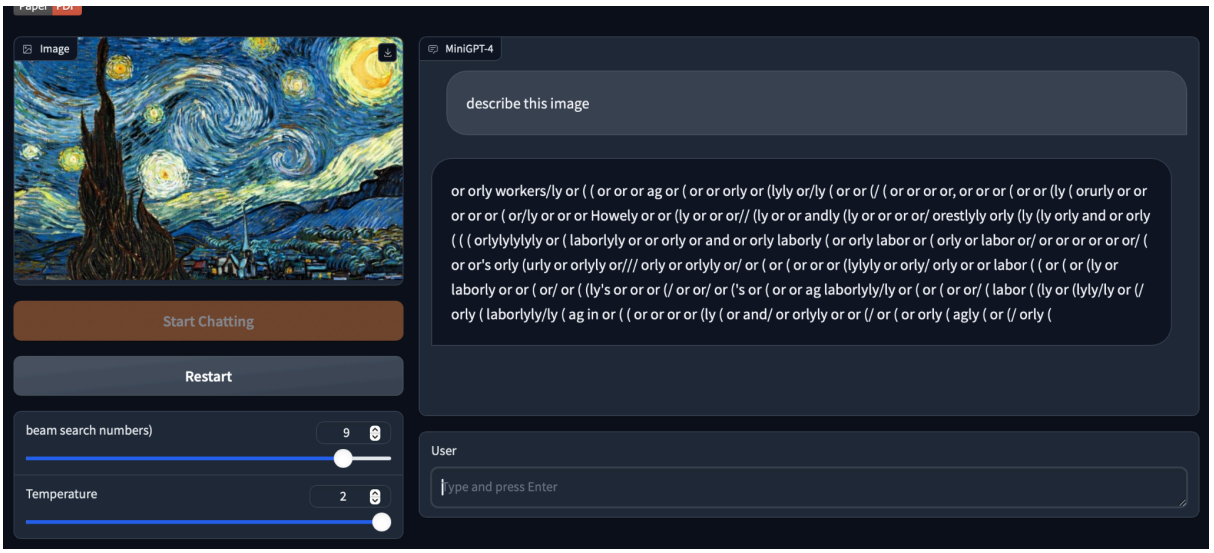
As GPUs que tenho acesso ficaram muito tempo fora do ar, o que dificultou bastante. Iniciei meus testes no Google Colab, mas possui algumas limitações como erros de execução, configuração de env e interrupções.

Quando a GPU 4090 voltou ao ar, primeiro conferi o armazenamento, apaguei alguns arquivos de checkpoints e principalmente de cache.

Iniciei o tutorial do próprio tutorial do github, porém o tutorial não fornece tantos detalhes e MUITAS coisas tive que corrigir manualmente na base de tentativa, erro e debugs. (O que gastou muitas horas).

Dentro do meu container, criei um env.
Clonei o repositório.
Instalei as dependências.
Baixei o modelo Phi2 (o que pode ser uma mudança que posso implementar futuramente)
Alterei alguns paths dentro dos arquivos de codigo para apontar para o caminho do Phi2.
Baixei os checkpoints do TinyGPT-V disponibilizando por eles (para teste incial).
Eles liberam em 4 estágios, testei com 3 deles (2, 3 e 4), que envolvem diferentes mudanças nos códigos.
Alterei mais alguns arquivos de códigos, inclusive alguns demorei muito tempo entendendo o que e onde deveria mudar (na base de tentativa e erro).
Adaptei o Phi2 ao Transformer (algumas modificações de biblioteca ocorreram desde o lançamento do modelo), o que também exigiu um bom trabalho.
Consegui rodar uma demo local com uso do Gradio.
Após muitas tentativas, finalmente consegui rodar.
Porém…





Não tive bons resultados.
Troquei os checkpoints, alterei mais algumas coisas e não consegui descobrir o problema.
Gastei também um bom tempo para corrigir.

Então decidi partir para a parte de train direto, para ver se tinha algum progresso e se com o meu treino esse problema era resolvido.

Iniciei a preparação dos dados no formato específico, essa é uma parte que também gera esforço já que cada um deve estar em um formato diferente.

Então iniciei as mudanças nos códigos, descomentei algumas linhas, mudei alguns caminhos de modelos e dados. E tentei iniciar o treinamento.

Porém ainda encontrei alguns erros de execução, além de uma loss zerada.

Então, ainda não consegui treinar de fato. Porém, tive um grande avanço.

APÊNDICE 5

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 5 de nov. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Tema: Modelos Multimodais

Durante o processo de Residência, explorei e estudei muitos papers, buscando cobrir trabalhos de diferentes:

- períodos,
- arquiteturas,
- motivações
- e aplicações.

Estudei desde surveys sobre datasets e benchmarks até modelos de compreensão e geração, incluindo arquiteturas autoregressivas e de difusão. Analisei classificações variadas (por tipo de encoder/decoder, tokenização, abordagem autoregressiva ou difusiva, e nível de representação- pixel ou token) com o objetivo de compreender de forma ampla o campo.

Elaborei resumos, tabelas comparativas e um mapa mental conectando os principais conceitos.

Durante as últimas Semanas, além de continuar os estudos, iniciei a parte prática.

Dados meus estudos, decidi começar tentando reproduzir o **LLaVA**, buscando replicar os experimentos originais do paper antes de propor modificações.
No entanto, percebi que, com os recursos disponíveis no momento, esse treinamento seria inviável. Encontrei então uma alternativa mais acessível: o **Mini-LLaVA**, que propõe um processo de treinamento mais leve e eficiente.

Durante todo o percurso, mantive um registro de ideias e possíveis implementações, aplicações e pontos de vista.
Um ponto observado por mim foi a limitação imposta pelo alto custo computacional da área, a maioria dos trabalhos exige recursos financeiramente inviáveis.

Na última Semana, decidi voltar novamente as minhas pesquisas de papers para definir o que eu

implementaria e decidi pelo: TinyGPT-V: Efficient Multimodal Large Language Model via Small Backbones -> TinyGPT-V: Efficient Multimodal Large Language Model via Small Backbones

Com isso, iniciei a implementação desse trabalho, clonei o repositório, rodei inferências com modelo local, fiz pequenas alterações. Iniciei o script de treino, porém eu ainda não estava satisfeita com o rumo do trabalho, faria um treinamento e depois não sabia exatamente o que faria.

Até este momento ainda estava considerando usar uma GPU 4090 - 24gb VRAM.

Decidi novamente, voltar para as pesquisas. Uma ideia que estava como plano B era implementar um **Retrieval Multimodal**:

- Essa linha utiliza arquiteturas como CLIP e SigLIP porém aplicados de forma distinta, voltada à recuperação de informações multimodais.

Assim, dediquei alguns dias à pesquisa de retrieval, analisando requisitos e viabilidade. Percebi que, para avançar em qualquer direção, precisaria expandir minha capacidade computacional.
Investiguei diversas alternativas, como o Colab Pro e outras plataformas, e identifiquei que a Vast.ai oferecia os melhores benefícios.

A partir disso, consolidei meus principais achados e iniciei a fase prática:

1. Execução de baselines: rodei alguns modelos em tarefas de retrieval multimodal, utilizando datasets menores e métricas de desempenho como: Recall@k, MRR e nDCG.
    a. Fiz o teste com Datasets em Inglês, Português e um de domínio específico - Artes.
    b. Código do Google Colab: testes_retriever.ipynb
    c. Resultados: Resultados - Baseline

2. Seleção de paper para reprodução PAPERS - Retrieval : escolhi o trabalho "Efficient Multimodal Retrieval with SigLIP", que não disponibiliza código, mas possui uma descrição suficientemente detalhada para permitir a replicação dos experimentos.
    a. Esse trabalho propõe uma arquitetura de **re-ranking** para melhorar o desempenho de modelos multimodais pré-treinados como CLIP, SigLIP, SigLIP-2 e BLIP-2 em tarefas de text-to-image retrieval.
    b. Optei por focar no SigLip

3. Implementação e adaptação: iniciei a implementação com base na metodologia descrita, registrando o processo no repositório github.com/juliadollis/residencia.

4. Redução de escopo: como o artigo original utiliza datasets com milhões de amostras (computacionalmente inviável para minha realidade), adaptei os experimentos para conjuntos menores.

5. Execução: inicialmente utilizei uma GPU A100 (80 GB de VRAM), mas, devido ao alto custo,

optei por uma GPU 5090 (32 GB de VRAM). Consegui reproduzir o treinamento, com muitos testes e ajustes.
  - Detalhes do trabalho: Experimentos ELIP (Como é a arquitetura, como implementei, etc)
  - Utilizei datasets de domínios gerais e específicos, de diferentes tamanhos.
  - Datasets que usei e que pretendo usar: Datasets - Retrieval

- Iniciei também a implementação de um Fine Tuning do SigLip para comparação. Resultados - Fine Tuning

Resultados:
- Por ainda estar adaptando o código, corrigindo erros, otimizando o treinamento e estar utilizando uma quantidade de dados muito menor que o paper, ainda não obtive resultados muito positivos.

Planejamento: [descrever o que pretende fazer para realizar a próxima ENTREGA]

- Melhorar o treinamento do ELIP, experimentações de:
  - Hiperparâmetros
  - Datasets (Tipo e Quantidade)
  - Épocas
  - Técnicas
  - Modelos (CLIP, etc)

- Comparações - Retrieval Geral e em Domínio:

Modelo x ELIP x Fine Tuning x Fine Tuning LoRA x Outros Trabalhos (https://arxiv.org/pdf/2505.21549 - https://arxiv.org/pdf/2411.00988v1)

Considerar custo, número de amostras e tamanho do modelo.

Possível implementação de RAG (Retrieval-Augmented Generation) - com modelo Generativo.

Observação: [caso precise fazer alguma observação, de qualquer "natureza"]

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Resultados Preliminares do Fine-tuning no Modelo SigLip

"lmms-lab/flickr30k", split="test[:1000]"

Modelo	Recall@1	Recall@5	Recall@10	MRR	nDCG@10
openai/clip-vit-large-patch14	0.6554	0.884	0.933	0.7540	0.7978
openai/clip-vit-base-patch16	0.6142	0.8648	0.9226	0.7192	0.7687
laion/CLIP-ViT-B-32-laion2B-s34B-b79K	0.6796	0.8956	0.9372	0.7717	0.8122
laion/CLIP-ViT-H-14-laion2B-s32B-b79K	0.7906	0.9468	0.9712	0.8600	0.8875
google/siglip-so400m-patch14-384	0.6306	0.8358	0.8930	0.7191	0.7611
google/siglip-base-patch16-224	0.2950	0.4990	0.5826	0.3817	0.4295

"lmms-lab/COCO-Caption", split="val[:1000]"

Modelo	Recall@1	Recall@5	Recall@10	MRR	nDCG@10
openai/clip-vit-large-patch14	0.4811	0.7282	0.8055	0.5854	0.6384
openai/clip-vit-base-patch16	0.4552	0.6986	0.7845	0.5602	0.6140
laion/CLIP-ViT-B-32-laion2B-s34B-b79K	0.4979	0.7535	0.8380	0.6079	0.6633

laion/CLIP-ViT-H-14-laion2B-s32B-b79K	0.5970	0.8183	0.8854	0.6921	0.7388
google/siglip-so400m-patch14-384	0.4100	0.6199	0.6994	0.5003	0.5480
google/siglip-base-patch16-224	0.0781	0.1809	0.2437	0.1226	0.1510

Artificio/WikiArt

Modelo	Recall@1	Recall@5	Recall@10	MRR	nDCG@10
openai/clip-vit-large-patch14	0.0009	0.0046	0.0095	0.0027	0.0043
openai/clip-vit-base-patch16	0.0015	0.0053	0.0102	0.0034	0.0049
laion/CLIP-ViT-B-32-laion2B-s34B-b79K	0.0009	0.0049	0.0094	0.0027	0.0043
laion/CLIP-ViT-H-14-laion2B-s32B-b79K	0.0010	0.0051	0.0102	0.0030	0.0046
google/siglip-so400m-patch14-384	0.0009	0.0042	0.0089	0.0026	0.0040
google/siglip-base-patch16-224	0.0008	0.0058	0.0118	0.0032	0.0051

Levantamento Bibliográfico de Trabalhos sobre Retrieval Imagem - Texto

Papers levantados como possíveis implementações:

- https://arxiv.org/pdf/2502.15682v2
- https://arxiv.org/pdf/2505.21549
- https://arxiv.org/pdf/2411.00988v1

- https://aclanthology.org/2024.emnlp-main.305.pdf
- https://aclanthology.org/2024.acl-long.783.pdf
- https://www.artemisdataset-v2.org
- https://www.artemisdataset.org
- https://github.com/NUS-HPC-AI-Lab/Multimodal-ICL-Retriever
- https://arxiv.org/html/2505.15877v1
- https://www.ecva.net/papers/eccv_2024/papers_ECCV/papers/11927.pdf
- https://arxiv.org/html/2501.04652v1
- https://aclanthology.org/2024.emnlp-main.1241.pdf
- https://arxiv.org/pdf/2505.07166
- https://www.semanticscholar.org/paper/Fine-Tuning-LLaMA-for-Multi-Stage-Text-Retrieval-Ma-Wang/a531e9a328ad1488567fa68c15d5bf30bfb90c78
- https://arxiv.org/pdf/2303.13220
- https://arxiv.org/pdf/2411.02571
- https://arxiv.org/html/2509.14985v1
- https://aclanthology.org/2024.acl-long.289.pdf - POKEMON
- https://arxiv.org/pdf/2412.14475
- https://arxiv.org/html/2504.01916v1
- https://arxiv.org/pdf/2302.06605
- https://arxiv.org/pdf/2210.04183
- https://arxiv.org/pdf/2209.13764
- https://openreview.net/forum?id=TE0KOzWYAF
- https://arxiv.org/pdf/2410.02069
- https://huggingface.co/prithivMLmods/siglip2-mini-explicit-content
- https://github.com/merveenoyan/siglip
- https://github.com/mlfoundations/open_clip/discussions/972
- https://huggingface.co/docs/transformers/v4.42.0/model_doc/siglip
- https://engineering.mercari.com/en/blog/entry/20241104-similar-looks-recommendation-via-vision-language-model/
- https://medium.com/kx-systems/guide-to-multimodal-rag-for-images-and-text-10dab36e3117
- https://github.com/mlfoundations/open_clip/tree/main

Experimentos com ELIP

ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval (https://arxiv.org/pdf/2502.15682v2)

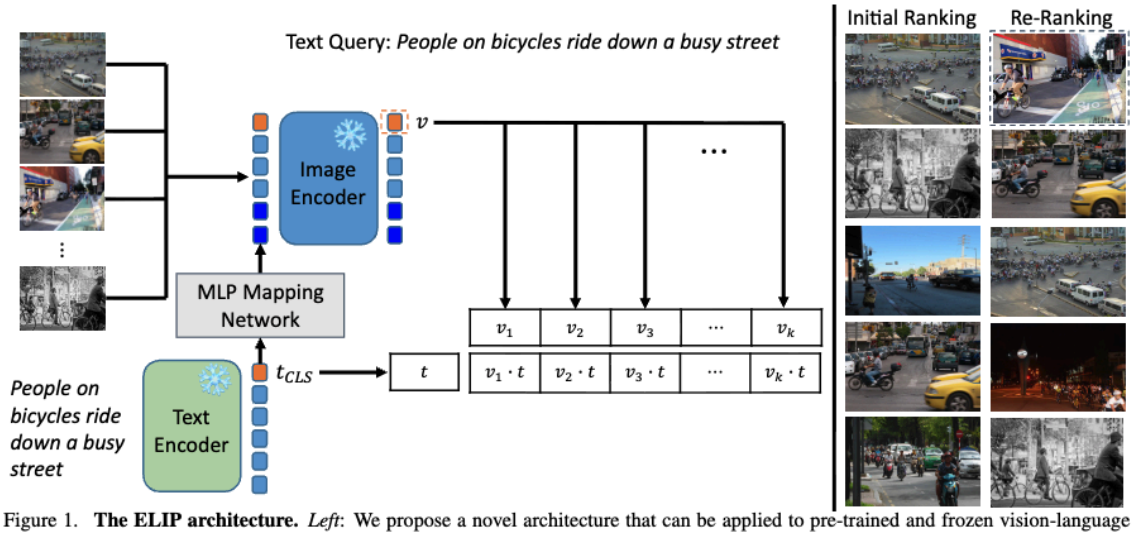


Figure 1. **The ELIP architecture.** *Left*: We propose a novel architecture that can be applied to pre-trained and frozen vision-language foundation models, such as CLIP, SigLIP, SigLIP-2 and BLIP-2, to enhance their text-to-image retrieval performance. The *key idea* is to use the text query to define a set of visual prompt vectors that are incorporated into the image encoder to make it aware of the query when generating the embedding. An MLP maps from the text space to the visual space of the input to the ViT encoder. The architecture is lightweight, and our data curation strategies enable efficient and effective training with limited resources. *Right*: In this retrieval example from the COCO benchmark, the top-k (k=100) images are re-ranked by our ELIP model for the text query: 'People on bicycles ride down a busy street'. The ground truth image matching the query is not in the top-5 ranked images in the initial CLIP ranking, but is ranked top-1 (highlighted in the dashed box) by the re-ranking.

ELIP: Mapeamento MLP que transforma o embedding do texto em vetores de prompt visuais -> são inseridos no codificador de imagem (ViT) para torná-lo *condicionado ao texto*.
O embedding visual é recalculado levando em conta a consulta textual, o que permite refinar o ranqueamento inicial de imagens gerado pelo modelo base.
Na inferência, o ELIP recalcula as features apenas para as *top-k* imagens do ranking inicial e refaz a ordenação.

RESULTADOS - DOMÍNIO ARTES
Treinamentos feitos com 1000 dados, 5000 dados e 103000 dados

1 época, 5 épocas, 10 épocas

Resultados 1 época:
>> Limitando avaliação a 60 amostras
>> Avaliando baseline
{'Baseline R@1': 0.0167, 'Baseline R@5': 0.0833, 'Baseline R@10': 0.2167, 'Baseline R@50': 0.8833}

{'Baseline mAP': 0.0876}
>> Avaliando ELIP-S
{'ELIP R@1': 0.0167, 'ELIP R@5': 0.1, 'ELIP R@10': 0.1667, 'ELIP R@50': 0.8667}
{'ELIP mAP': 0.0844}

100 AMOSTRAS:

Baseline imagens:
{'Baseline R@1': 0.01, 'Baseline R@5': 0.06, 'Baseline R@10': 0.11, 'Baseline R@50': 0.6}
{'Baseline mAP': 0.0591}
>> Avaliando ELIP-S
{'ELIP R@1': 0.01, 'ELIP R@5': 0.05, 'ELIP R@10': 0.09, 'ELIP R@50': 0.52}
{'ELIP mAP': 0.0534}

INGLES - DF GERAL
{'Baseline R@1': 0.104, 'Baseline R@5': 0.226, 'Baseline R@10': 0.322, 'Baseline R@50': 0.626}
{'Baseline mAP': 0.1754}

{'ELIP R@1': 0.002, 'ELIP R@5': 0.006, 'ELIP R@10': 0.014, 'ELIP R@50': 0.104}
{'ELIP mAP': 0.0118}

Datasets para Retrieval

Lista de levantamento de possíveis datasets à serem usados no Fine-Tuning dos modelos CLIP e SigLip.

Geral - Inglês

- https://github.com/luomancs/ReMuQ -> Retriever
- https://github.com/google-research-datasets/wit -> Wikipedia Retriever
- https://huggingface.co/datasets/TIGER-Lab/M-BEIR
- https://huggingface.co/datasets/pixparse/cc3m-wds
- https://huggingface.co/datasets/apple/DataCompDR-12M
- https://huggingface.co/datasets/lmms-lab/flickr30k
- https://huggingface.co/datasets/Mumon/COCO_Captions

- https://huggingface.co/datasets/lmms-lab/COCO-Caption2017

Arte:

- https://huggingface.co/datasets/Artificio/WikiArt
- https://www.kaggle.com/datasets/samamostafa03/artemis-dataset
- https://www.artemisdataset-v2.org (https://drive.google.com/file/d/11UW0DyhTuUkNh2Mm1i_NRVR8nwscq9Ph/edit)
- https://noagarcia.github.io/SemArt/
- https://arxiv.org/pdf/2509.14891
- https://arxiv.org/html/2507.21917v1
- https://drops.dagstuhl.de/storage/08tgdk/tgdk-vol002/tgdk-vol002-issue002/TGDK.2.2.8/TGDK.2.2.8.pdf
- https://arxiv.org/pdf/2503.17116
- https://www.kaggle.com/datasets/ziya07/multimodal-artistic-style-conversion-dataset
- https://arxiv.org/pdf/2503.17116

PT-BR

- https://huggingface.co/datasets/laicsiifes/coco-captions-pt-br
- https://huggingface.co/datasets/VerboVision/PraCegoVer-Filtrado-FSB
- https://huggingface.co/datasets?search=LAICSI-IFES

Resultados Fine-tuning no Domínio Artístico

DOMÍNIO ARTES NO DATASET Artificio/WikiArt

Modelos sem qualquer ajuste/fine-tuning:

Modelo	Recall@1	Recall@5	Recall@10	MRR	nDCG@10
openai/clip-vit-large-patch14	0.0009	0.0046	0.0095	0.0027	0.0043
openai/clip-vit-base-patch16	0.0015	0.0053	0.0102	0.0034	0.0049
laion/CLIP-ViT-B-32-laion2B-s34B-b79K	0.0009	0.0049	0.0094	0.0027	0.0043
laion/CLIP-ViT-H-14-laion2B-s32B-b79K	0.0010	0.0051	0.0102	0.0030	0.0046
google/siglip-so400m-patch14-384	0.0009	0.0042	0.0089	0.0026	0.0040
google/siglip-base-patch16-224	0.0008	0.0058	0.0118	0.0032	0.0051

Modelos com Fine-tuning:

{'recall@1': 0.0010339149636853324, 'recall@5': 0.005118517289355782, 'recall@10': 0.009637108612128716, 'mrr': 0.0029748963096549986, 'ndcg@10': 0.004502982941548234}

{'recall@1': 0.0009445642878112914, 'recall@5': 0.005412098081513345,
'recall@10': 0.010875253692097572, 'mrr': 0.0030997289290946666,
'ndcg@10': 0.004878210654330872}

{'recall@1': 0.0009317999055435712, 'recall@5': 0.005361040552442464,
'recall@10': 0.010211505814176123, 'mrr': 0.0030425576661360326,
'ndcg@10': 0.004687965797952993}

APÊNDICE 6

Termo de Aceite de Entrega

Objetivo deste documento

Este documento faz parte do Processo da disciplina Residência em IA e tem como objetivo formalizar o aceite da entrega considerando o planejado e o realizado para o período.

Data da Reunião ("Gate") de aprovação: 13 de nov. de 2025

Participantes da Entrega [matriculados em Residência em IA]:

JULIA SOARES DOLLIS

Entrega: [descrever a ENTREGA - requisitos e produtos gerados: links para textos, códigos, vídeos etc.]

Tema: Modelos Multimodais

- Estudei o tema de forma ampla e geral
- Me aprofundei em Retrieval Multimodal Texto - Imagem

Durante esta última Semana:

- Dei continuidade a implementação do **ELIP**
- Desafios:
  - Para replicar o paper são necessários mais de 1T de armazenamento (O paper se diz "student friendly") com o dataset apple/DataCompDR-12M · Datasets at Hugging Face
  - Mesmo para o menor dataset experimentado no paper (WeiChow/cc3m · Datasets at Hugging Face) ainda são necessários mais de 600GB de armazenamento.
  - Por se tratar de uma MLP simples, são necessários muitos dados para que ela de fato aprenda algo.
- Resultados:
  - Dado esses desafios, "repliquei"o que foi proposto no paper, desenvolvendo o código por conta própria já que não disponibilizam.
  - Experimentos:
    - Muitos testes pequenos para ajustes do código e conferência.
    - Repliquei o uso do Dataset cc3m porém com apenas 1000 dados.

- Utilizei o Dataset FLICKR30K-PTBR (https://huggingface.co/datasets/laicsiifes/flickr30k-pt-br) que contém 30 mil amostras.
- Em outro treino, utilizei o Dataset WikiArt (https://huggingface.co/datasets/Artificio/WikiArt) que contém 103 mil amostras.
- Resultados - ELIP - Resultados do Paper comp

FLICKR30K-PTBR

Baseline: 0.4020 | 0.6930 | 0.7830 | 0.9250 | 0.5357 | 0.5890
Rerank: 0.0130 | 0.0490 | 0.1060 | 0.5560 | 0.0558 | 0.0485

WIKIART

Baseline: 0.6240 | 0.8340 | 0.8720 | 0.9460 | 0.7146 | 0.7493
Rerank: 0.0640 | 0.1720 | 0.2580 | 0.7760 | 0.1371 | 0.1453

- Aprimorei a imagem/mapa mental que eu havia feito anteriormente, adicionando novas informações sobre Retrieval. imagem.pdf

- Gerei Baselines completos dos modelos CLIP [clip-vit-base-patch16, clip-vit-large-patch14] e SigLip [siglip-so400m-patch14-384, google/siglip-base-patch16-224] nos Datasets [coco-captions, flickr8k e flickr30k] em Português e em Inglês.

- Assim pude avaliar o desempenho dos modelos base, sem nenhum tipo de adaptação e suas diferenças em Português e em Inglês.
  - O modelo SigLip se desempenha melhor no geral e em PT-BR. Ele é Multillingue.

- Além disso, também testei no domínio de obras artísticas [WikiArt, Moda]
- Outros como: Pokemon, Exames médicos, outros Datasets de Arte e Moda.

- Construí códigos de Fine Tuning do CLIP e SigLip, e de inferência.

- Foram realizados Fine Tunings diversos (Muitos para testes, principalmente de domínios, os resultados não constam na tabela final de Resultados)
- Análise do tempo e custo das abordagens.

Resultados Fine Tuning:

- Resultados Fine Tuning

Principais Resultados:

DeepFashion (domínio de moda)

Modelo: openai/clip-vit-base-patch16
recall@1: 0.00870 | ndcg@10: 0.02377

Modelo: openai/clip-vit-large-patch14
recall@1: 0.00823 | ndcg@10: 0.02876

Modelo: google/siglip-base-patch16-224
recall@1: 0.02492 | ndcg@10: 0.06460

Modelo: turing552/siglip-fashion-5ep
recall@1: 0.31100 | ndcg@10: 0.53051

Modelo: turing552/clip-deepfashion-multimodal-10ep
recall@1: 0.31852 | ndcg@10: 0.58128

WikiArt —

Modelo: google/siglip-base-patch16-224
recall@1: 0.18605

Modelo: google/siglip-so400m-patch14-384

recall@1: 0.39128

Modelo: turing552/siglipbase-wikiart-5ep
recall@1: 0.30906

Modelo: turing552/clip-wikiart-raw-v1-10ep
recall@1: 0.43826

Adaptação para Português (CLIP Base vs Fine-tuned)

Flickr30k-pt-br —

Modelo: openai/clip-vit-large-patch14
recall@1: 0.18310

Modelo: turing552/cliplarge-flickr30k-ptbr-5ep-novo
recall@1: 0.76310

SigLIP

Flickr30k-pt-br — EM LINHAS

Modelo: google/siglip-so400m-patch14-384
recall@1: 0.50655

Modelo: turing552/siglip-flickr30k-ptbr-5ep
recall@1: 0.80448

Tempo e Custo

ELIP muitos dados — 18h — 9.00 USD
ELIP 100k dados — 5h — 2.50 USD
Fine-tuning 100k — 1h30min — 0.75 USD
Fine-tuning (˜30K) — 30min — 0.25 USD
Sem fine-tuning — 0h — 0.00 USD

- GitHub com um Guia sobre Retrieval Multimodal Texto - Imagem com definições, dicas, passo e passo, papers sugeridos e códigos (com tutoriais). https://github.com/juliadollis/retrieval_image_text/tree/main
    - No repositório também estão os resultados de todos os meus testes dos modelos com resultado das métricas.

- Além do Repositório conter a parte teórica e de códigos brutos, criei demonstrações práticas que podem ser executadas com qualquer modelo e dataset. Assim, temos uma aplicação de fato para a teoria, que é completamente adaptável.

- **1. Demonstração de Métricas de Retrieval** em qualquer dado, gerando métricas e insights.
    - O usuário coloca o caminho do Dataset HuggingFace com split e nome das colunas
    - Inferencia de Retrieval em dois modelos CLIP e dois modelos SigLip
    - Print das métricas de Retrieval
    - Análises e Feedbacks sobre seu dataset e os resultados, dando dicas sobre o que fazer (usar o modelo, fazer fine tuning, etc)
    - Uso do Gradio
- **2. Demonstração de Retrieval** -
    - 2.1 - Busca de Roupas pela descrição
        - Aplicação simples de Retrieval que pode ser adaptada à qualquer Dataset de texto-imagem
        - Usuário coloca uma descrição (ex.:yellow shirt) -> Retrieval Query <-> Imagens -> 10 imagens mais similares são retornadas para o usuário

- Uso do modelo fine tunado [turing552/clip-wikiart-raw-v1-10ep]
- 2.2 - Busca de Obras de Arte pela descrição, tema, artista
  - Usuário coloca uma descrição (ex.:ballet, impressionism) -> Retrieval Query <-> Obras -> 10 obras mais similares são retornadas para o usuário

- **3. Demonstração/Aplicação de RAG**
  - Para de fato ter uma aplicação dos estudos de Retrieval e unir o conhecimento sobre VLMs (LLava)
  - Aplicação do Retrieval tem de fato uma utilidade vinculada à um modelo.
  - Dataset de Pokemon (Pode ser facilmente modificado - é apenas uma demonstração divertida)
  - Uso do Gradio
  - Usuário coloca a descrição do pokemon -> imagem é encontrada -> LLava recebe imagem, nome e informações do Pokemon e monta um texto criativo sobre ele como resposta ao usuário.
  - Demonstração Métricas - Exemplos das Demonstrações, Códigos e Detalhes
- No repositório do GitHub tem o link para o Demonstração de Métricas, do RAG e Retrieval Simples e também os códigos para serem executados em qualquer máquina (gerando um link local).

- Assim, além de explicações, tutoriais, papers, códigos e resultados reais, também tem partes extremamente práticas e visuais para a consolidação do conhecimento.

Links - Demonstrações
- 1 https://ee94b26a4eec384aea.gradio.live/
- 2.1 https://0e21131f0520121642.gradio.live/
- 2.2 https://28a1008c3d1c1030e5.gradio.live
- 3 https://a133636e14105ede19.gradio.live/

Custo Total: $120 × 5,4 = **648 reais**

Planejamento: **[descrever o que pretende fazer para realizar a próxima ENTREGA]**

Observação: **[caso precise fazer alguma observação, de qualquer "natureza"]**

Acho que meu principal aprendizado é que eu consigo de fato "resolver um problema" sozinha. Mesmo quando tudo deu errado (muitas e muitas vezes), consegui recalcular os planos, rever o que estava errado e recomeçar. Além de me sentir mais capaz de FAZER ne sinto mais capaz de PLANEJAR.
Além de Modelos Multimodais (difusão incluso), aprendi muito sobre a importância de calcular os requisitos antes do início de um projeto.

ACEITE DA ENTREGA:

CEDRIC LUIZ DE CARVALHO: Go!

Resultados ELIP

Comparativos dos resultados reportados no Paper vs Resultados obtidos nos meus experimentos.

TABELA 1 – ABLATION (ELIP) Fonte: arXiv:2502.15682v2 [cs.CV] 27 Mar 2025

Setting	Architecture	Training Dataset	Hard Sample Mining	Multiple Prompts	COCO R@1	COCO R@5	COCO R@10	COCO Avg	Flickr R@1	Flickr R@5	Flickr R@10	Flickr Avg
A	CLIP	–	–	–	40.2	66.0	75.6	60.6	67.6	88.3	93.0	83.0
B	ELIP-C	CC3M	–	–	40.7	66.2	76.1	60.7	68.8	89.3	93.8	83.8
C	ELIP-C	CC3M	✓	–	41.8	67.5	77.5	62.3	69.5	89.7	94.1	84.4
D	ELIP-C	DataCompDR	–	–	44.2	70.0	79.5	64.6	71.3	90.6	94.4	85.4
E	ELIP-C	DataCompDR	✓	✓	45.6	71.1	80.4	65.7	72.3	90.6	94.7	85.9

TABELA 2 – STATE OF THE ART Fonte: arXiv:2502.15682v2 [cs.CV] 27 Mar 2025

Modelo	Ano	COCO R@1	COCO R@5	COCO R@10	COCO Avg	Flickr R@1	Flickr R@5	Flickr R@10	Flickr Avg

CLIP	2021	40.16	65.95	75.62	60.58	67.56	88.34	93.00	82.97
ELIP-C (Ours)	–	45.61	71.08	80.43	65.71	72.30	90.62	94.68	85.87
SigLIP	2023	54.21	76.78	84.24	71.74	82.96	96.10	98.04	92.37
ELIP-S (Ours)	–	61.03	82.62	88.70	77.45	87.62	98.16	99.16	94.98
SigLIP-2	2025	56.87	78.79	85.49	73.72	83.94	96.62	98.20	92.92
ELIP-S2 (Ours)	–	62.91	83.86	89.70	78.82	87.74	97.96	98.94	94.88
BLIP-2	2023	68.25	87.72	92.63	82.87	89.74	98.18	98.94	95.62
Q-Pert. (E*)	2024	68.34	87.76	92.63	82.91	89.82	98.20	99.06	95.71
Q-Pert. (D*)	2024	68.35	87.72	92.65	82.93	89.86	98.20	99.06	95.71
ELIP-B (Ours)*	–	68.41	87.88	92.78	83.02	90.08	98.34	99.22	95.88

TABELA 3 – FLICKR30K-PTBR

Método	**R@1**	**R@5**	**R@10**	**R@50**	**MRR**	**nDCG@10**

Baseline	0.4020	0.6930	0.7830	0.9250	0.5357	0.5890
Rerank	0.0130	0.0490	0.1060	0.5560	0.0558	0.0485

TABELA 4 – WIKIART

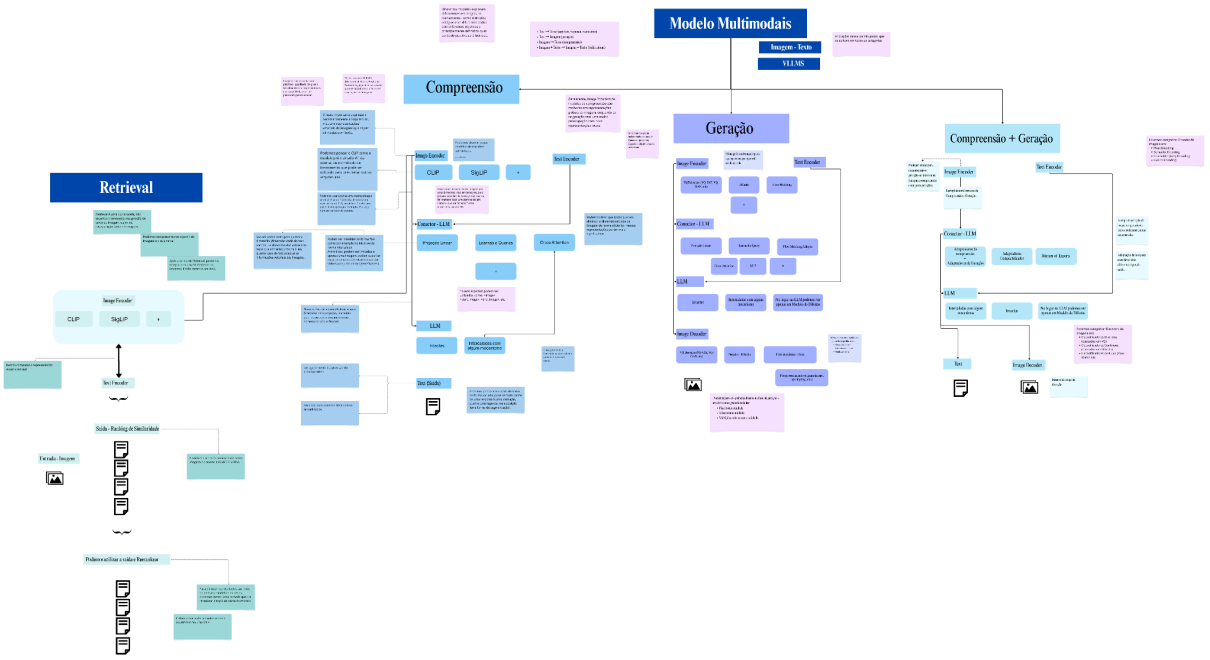
Método	**R@1**	**R@5**	**R@10**	**R@50**	**MRR**	**nDCG@10**
Baseline	0.6240	0.8340	0.8720	0.9460	0.7146	0.7493
Rerank	0.0640	0.1720	0.2580	0.7760	0.1371	0.1453

TABELA 5 – CC3M (1000 AMOSTRAS)

Modelo	**R@1**	**R@5**	**R@10**	**MRR**	**nDCG@10**
CC3M-1000	0.0001785	0.0010462	0.0019160	0.0005713	0.0008797

Mapa Mental completo de Modelos Multimodais

Mapa mental atualizado com informações sobre Retrieval.



Para melhor resolução:
https://www.canva.com/design/DAG4il1xBno/Oi3JdbGM_cohI3LDV9H3jg/edit?utm_content=DAG4il1xBno&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

Resultados Fine Tuning

Fine Tuning dos modelos CLIP e SigLip.

Diversos datasets.

TABELAS POR MODELO

Modelo: turing552/cliplarge-flickr30k-ptbr-5ep-novo

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
laicsiifes/coco-captions-pt-br	0.39799	0.64645	0.74345	0.50406	0.56125

laicsiifes/flickr8k-pt-br	0.83333	0.97333	0.99000	0.89059	0.91513
laicsiifes/flickr30k-pt-br	0.76310	0.93862	0.96897	0.83823	0.87036
lmms-lab/flickr30k	0.91538	0.99528	0.99780	0.94882	0.96114
Artificio/WikiArt	0.22702	0.44436	0.54015	0.31920	0.37170

Modelo: openai/clip-vit-base-patch16

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
laicsiifes/coco-captions-pt-br	0.06982	0.16814	0.23469	0.11321	0.14166
laicsiifes/flickr8k-pt-br	0.21833	0.43500	0.56500	0.31519	0.37386
laicsiifes/flickr30k-pt-br	0.08897	0.22000	0.30000	0.14599	0.18215
lmms-lab/flickr30k	0.70494	0.90815	0.94998	0.79154	0.83026
Artificio/WikiArt	0.19903	0.41705	0.52407	0.29292	0.34768
Marqo/deepfashion-multimodal	0.00870	0.02774	0.04513	0.01736	0.02377

Modelo: openai/clip-vit-large-patch14

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
laicsiifes/coco-captions-pt-br	0.11354	0.25522	0.34509	0.17651	0.21607
laicsiifes/flickr8k-pt-br	0.29333	0.58000	0.71000	0.41426	0.48445
laicsiifes/flickr30k-pt-br	0.18310	0.35897	0.45552	0.26025	0.30633
lmms-lab/flickr30k	0.71312	0.91696	0.95974	0.80018	0.83918
Artificio/WikiArt	0.26741	0.52019	0.62470	0.37466	0.43425

Marqo/deepfashion-multimodal	0.00823	0.03479	0.05712	0.02022	0.02876

Modelo: google/siglip-so400m-patch14-384

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
laicsiifes/coco-captions-pt-br	0.36345	0.61106	0.71144	0.46885	0.52672
laicsiifes/flickr8k-pt-br	0.68333	0.90667	0.94333	0.77672	0.81764
laicsiifes/flickr30k-pt-br	0.50655	0.75172	0.81897	0.61043	0.66084
lmms-lab/flickr30k	0.91979	0.98930	0.99654	0.95038	0.96192
Artificio/WikiArt (ATUALIZADO)	0.39128	0.64833	0.73801	0.49962	0.55678
Marqo/deepfashion-multimodal	0.02374	0.06911	0.11048	0.04400	0.05934

Modelo: turing552/siglip-flickr30k-ptbr-5ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
laicsiifes/coco-captions-pt-br	0.55152	0.78971	0.85699	0.65150	0.70114
laicsiifes/flickr8k-pt-br	0.89333	0.99333	0.99833	0.93835	0.95358
laicsiifes/flickr30k-pt-br	0.80448	0.95552	0.97828	0.86865	0.89568
lmms-lab/flickr30k	0.88833	0.98616	0.99434	0.93265	0.94820

Modelo: google/siglip-base-patch16-224

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
laicsiifes/coco-captions-pt-br	0.74600	0.95400	0.99200	0.83357	0.87238
laicsiifes/flickr8k-pt-br	0.93000	1.00000	1.00000	0.96167	0.97155
laicsiifes/flickr30k-pt-br	0.85000	0.99000	1.00000	0.91708	0.93817
lmms-lab/flickr30k	0.84303	0.96917	0.98364	0.89794	0.91926
Artificio/WikiArt (ATUALIZADO)	0.18605	0.37956	0.47002	0.26851	0.31627
Marqo/deepfashion-multimodal	0.02492	0.07593	0.11777	0.04849	0.06460

Modelo: turing552/siglip-wikiart-5ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
Artificio/WikiArt	0.30363	0.57734	0.69801	0.42128	0.48712

Modelo: turing552/siglipbase-wikiart-5ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
Artificio/WikiArt	0.30906	0.58412	0.69860	0.42561	0.49063

Modelo: turing552/clip-wikiart-raw-v1-5ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
Artificio/WikiArt	0.36736	0.71051	0.82625	0.51144	0.58699

Modelo: turing552/clip-wikiart-raw-v1-10ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
Artificio/WikiArt	0.43826	0.76407	0.85598	0.57683	0.64435

Modelo: turing552/clip-deepfashion-multimodal-10ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
Marqo/deepfashion-multimodal	0.31852	0.72661	0.87236	0.48960	0.58128

Modelo: turing552/siglip-fashion-5ep

Dataset	recall@1	recall@5	recall@10	mrr	ndcg@10
Marqo/deepfashion-multimodal	0.31100	0.64716	0.77927	0.45251	0.53051

TABELAS POR DATASET

Dataset: laicsiifes/coco-captions-pt-br

Modelo	recall@1	recall@5	recall@10	mrr	ndcg@10
turing552/cliplarge-flickr30k-ptbr-5ep-novo	0.39799	0.64645	0.74345	0.50406	0.56125
openai/clip-vit-base-patch16	0.06982	0.16814	0.23469	0.11321	0.14166
openai/clip-vit-large-patch14	0.11354	0.25522	0.34509	0.17651	0.21607
google/siglip-so400m-patch14-384	0.36345	0.61106	0.71144	0.46885	0.52672
turing552/siglip-flickr30k-ptbr-5ep	0.55152	0.78971	0.85699	0.65150	0.70114
google/siglip-base-patch16-224	0.74600	0.95400	0.99200	0.83357	0.87238

Dataset: laicsiifes/flickr8k-pt-br

Modelo	recall@1	recall@5	recall@10	mrr	ndcg@10
turing552/cliplarge-flickr30k-ptbr-5ep-novo	0.83333	0.97333	0.99000	0.89059	0.91513
openai/clip-vit-base-patch16	0.21833	0.43500	0.56500	0.31519	0.37386
openai/clip-vit-large-patch14	0.29333	0.58000	0.71000	0.41426	0.48445
google/siglip-so400m-patch14-384	0.68333	0.90667	0.94333	0.77672	0.81764

turing552/siglip-flickr30k-ptbr-5ep	0.89333	0.99333	0.99833	0.93835	0.95358
google/siglip-base-patch16-224	0.93000	1.00000	1.00000	0.96167	0.97155

Dataset: laicsiifes/flickr30k-pt-br

Modelo	recall@1	recall@5	recall@10	mrr	ndcg@10
turing552/cliplarge-flickr30k-ptbr-5ep-novo	0.76310	0.93862	0.96897	0.83823	0.87036
openai/clip-vit-base-patch16	0.08897	0.22000	0.30000	0.14599	0.18215
openai/clip-vit-large-patch14	0.18310	0.35897	0.45552	0.26025	0.30633
google/siglip-so400m-patch14-384	0.50655	0.75172	0.81897	0.61043	0.66084
turing552/siglip-flickr30k-ptbr-5ep	0.80448	0.95552	0.97828	0.86865	0.89568
google/siglip-base-patch16-224	0.85000	0.99000	1.00000	0.91708	0.93817

Dataset: lmms-lab/flickr30k

Modelo	recall@1	recall@5	recall@10	mrr	ndcg@10
turing552/cliplarge-flickr30k-ptbr-5ep-novo	0.91538	0.99528	0.99780	0.94882	0.96114
openai/clip-vit-base-patch16	0.70494	0.90815	0.94998	0.79154	0.83026

openai/clip-vit-large-patch14	0.71312	0.91696	0.95974	0.80018	0.83918
google/siglip-so400m-patch14-384	0.91979	0.98930	0.99654	0.95038	0.96192
turing552/siglip-flickr30k-ptbr-5ep	0.88833	0.98616	0.99434	0.93265	0.94820
google/siglip-base-patch16-224	0.84303	0.96917	0.98364	0.89794	0.91926

Dataset: Artificio/WikiArt

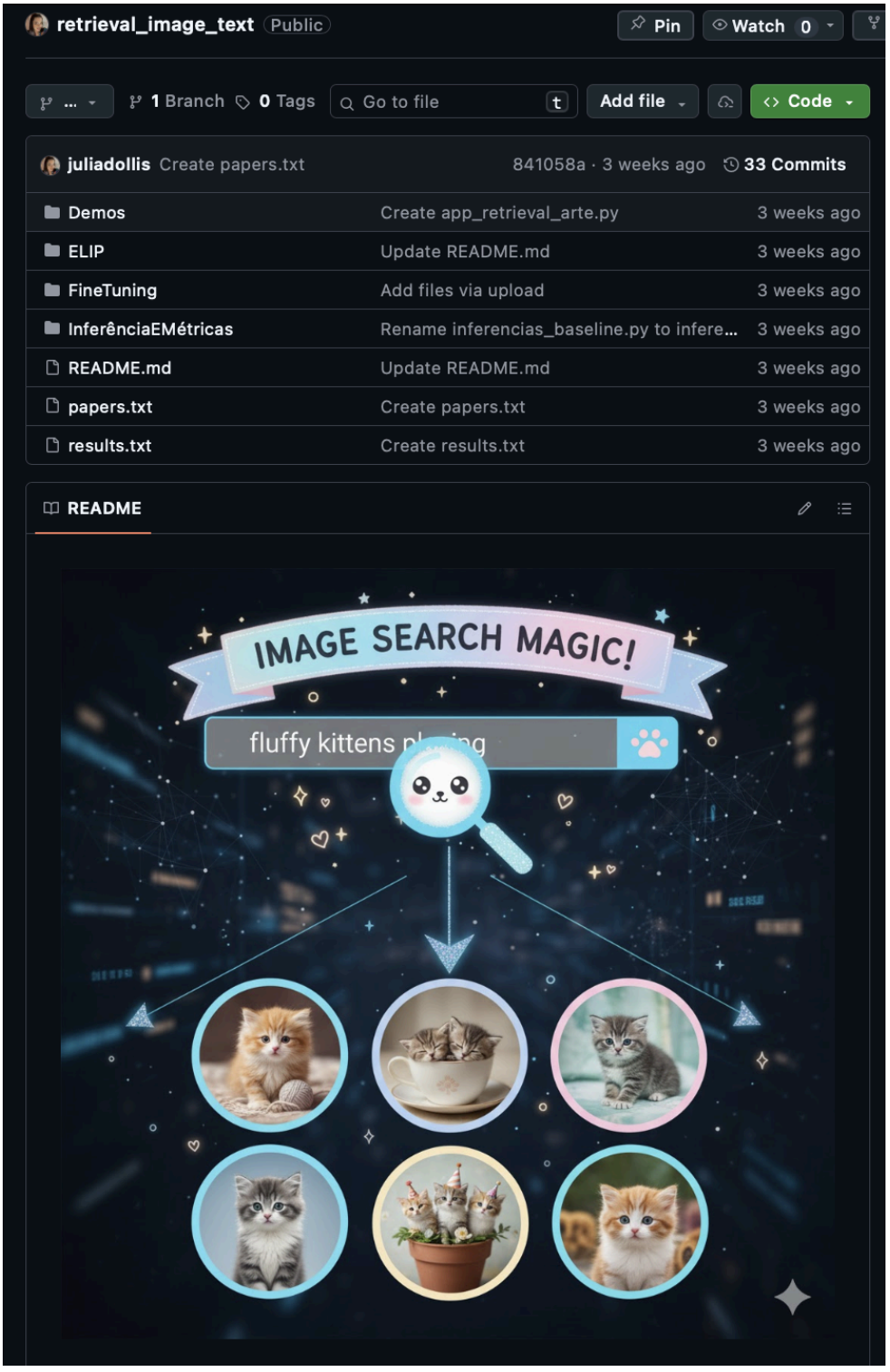
Modelo	recall@1	recall@5	recall@10	mrr	ndcg@10
openai/clip-vit-base-patch16	0.19903	0.41705	0.52407	0.29292	0.34768
openai/clip-vit-large-patch14	0.26741	0.52019	0.62470	0.37466	0.43425
google/siglip-so400m-patch14-384	0.29128	0.54833	0.63801	0.39962	0.45678
google/siglip-base-patch16-224)	0.18605	0.37956	0.47002	0.26851	0.31627
turing552/siglip-wikiart-5ep	0.30363	0.57734	0.69801	0.42128	0.48712
turing552/cliplarge-flickr30k-ptbr-5ep-novo	0.22702	0.44436	0.54015	0.31920	0.37170
turing552/siglipbase-wikiart-5ep	0.30906	0.58412	0.69860	0.42561	0.49063
turing552/clip-wikiart-raw-v1-5ep	0.36736	0.71051	0.82625	0.51144	0.58699

turing552/clip-wikiart-raw-v1-10ep	0.43826	0.76407	0.85598	0.57683	0.64435

Dataset: Marqo/deepfashion-multimodal

Modelo	recall@1	recall@5	recall@10	mrr	ndcg@10
openai/clip-vit-base-patch16	0.00870	0.02774	0.04513	0.01736	0.02377
openai/clip-vit-large-patch14	0.00823	0.03479	0.05712	0.02022	0.02876
google/siglip-so400m-patch14-384	0.02374	0.06911	0.11048	0.04400	0.05934
google/siglip-base-patch16-224	0.02492	0.07593	0.11777	0.04849	0.06460
turing552/clip-deepfashion-multimodal-10ep	0.31852	0.72661	0.87236	0.48960	0.58128
turing552/siglip-fashion-5ep (NOVO)	0.31100	0.64716	0.77927	0.45251	0.53051

Repositório GitHub



README

O que é Retrieval Imagem-Texto?

Retrieval imagem-texto é a tarefa de buscar imagens relevantes a partir de um texto (texto → imagem) ou buscar textos relevantes a partir de uma imagem (imagem → texto). A ideia central é que um modelo multimodal aprende a representar imagens e textos no mesmo espaço vetorial, onde itens semanticamente semelhantes ficam próximos.

Isso permite que uma consulta como "um cachorro correndo na praia" retorne automaticamente as imagens mais próximas desse conceito. Da mesma forma, fornecer uma imagem permite recuperar descrições ou legendas que melhor representam seu conteúdo.

Retrieval é uma das técnicas fundamentais de visão-linguagem e serve como base para diversos sistemas modernos, como RAG multimodal, motores de busca visuais, geração guiada por imagem e organização de grandes acervos multimodais.

Para que usamos Retrieval?

Busca por imagens Usado para encontrar rapidamente, dentro de milhares ou milhões de imagens, aquelas que são mais relevantes para um texto. Aplicações incluem mecanismos de busca, bancos de arte, fotografia, e-commerces e catálogos digitais.

Organização de acervos multimodais Auxilia na criação de índices, na recomendação visual e na detecção de similaridade.

RAG multimodal Em modelos que entendem imagens e textos, recuperar imagens relevantes melhora profundamente a qualidade das respostas. Exemplo: perguntar sobre uma obra de arte e recuperar quadros correlatos para análise.

Sistemas interativos Aplicações diversas, como buscar receitas pela foto, encontrar documentos escaneados semelhantes ou localizar produtos no varejo a partir de uma imagem.

Base para modelos generativos Muitos modelos usam retrieval para encontrar pares semelhantes, auxiliar no condicionamento ou melhorar datasets utilizados em pré-treinamentos.

Quais modelos podemos usar para retrieval e por quê?

Modelos CLIP (OpenAI) Simples, rápidos e amplamente usados. Boas opções para ensino e experimentação. Limitados em textos longos e idiomas fora do inglês.

Modelos CLIP da comunidade (OpenCLIP, LAION) openai/clip-vit-base-patch32 openai/clip-vit-large-patch14 laion/CLIP-ViT-H-14-laion2B Modelos maiores, treinados em datasets massivos, com desempenho superior ao CLIP original.

Modelos SigLIP (Google) google/siglip-so400m-patch14-384 google/siglip-base-patch16-224 Mais estáveis, geralmente melhores que CLIP e fortes em zero-shot.

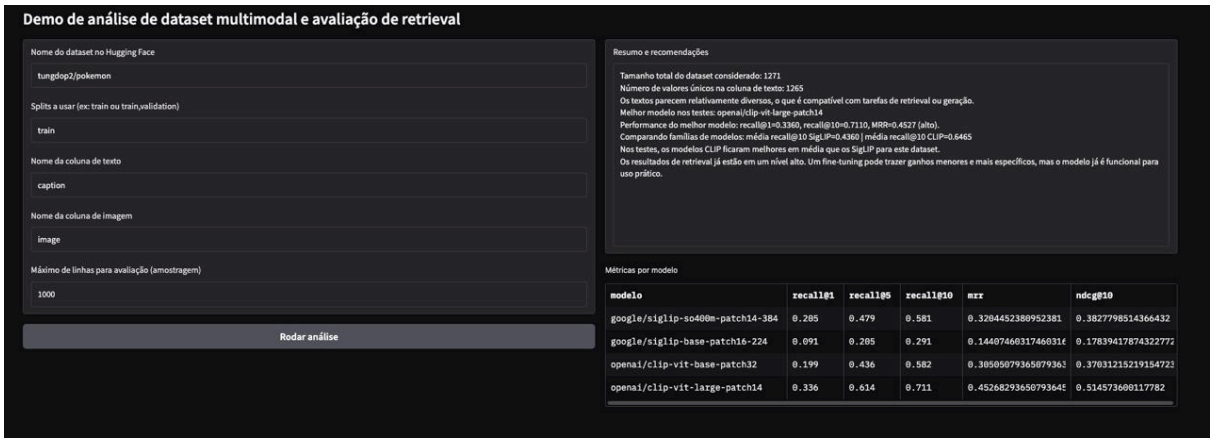
SigLIP 2 Suporte nativo a múltiplos idiomas, incluindo português. Desempenho excelente para tasks multilíngue.

Modelos fine-tunados em domínio específico Exemplos: flickr30k-pt-br-5ep, wikiart-ft Adaptam embeddings ao domínio, melhoram recall@K e superam facilmente modelos base

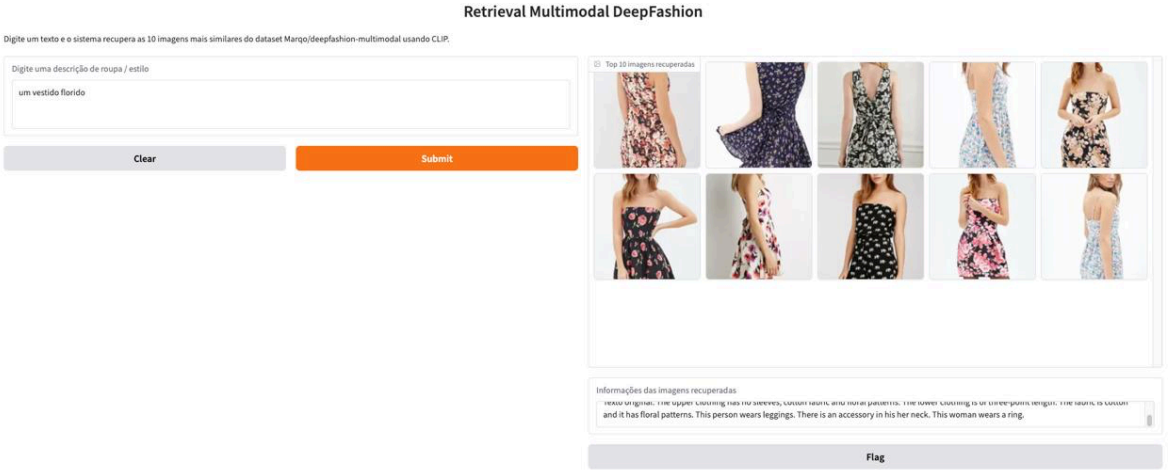
Para mais, acesse o repositório.

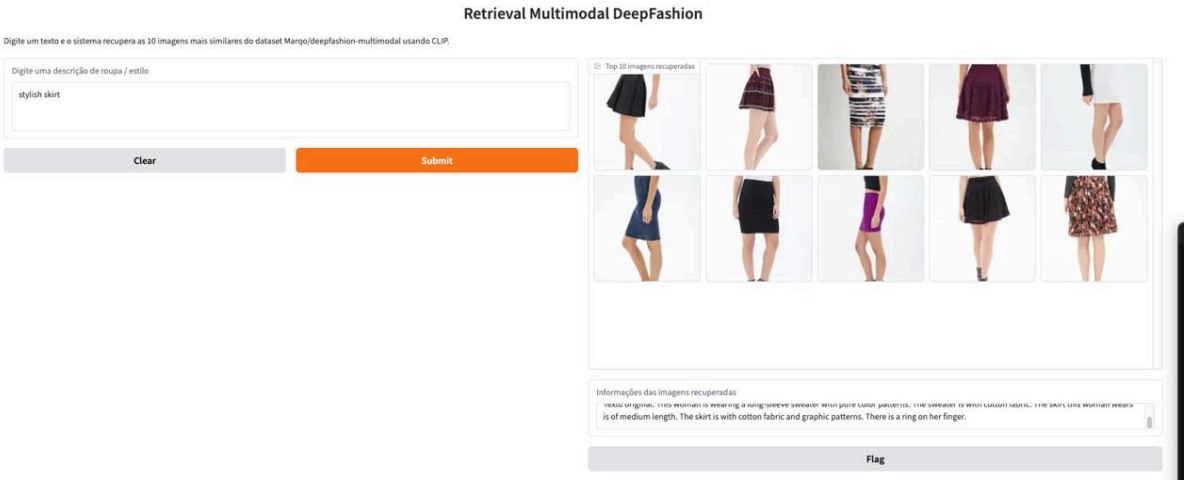
Exemplos e Código das Demonstrações desenvolvidas

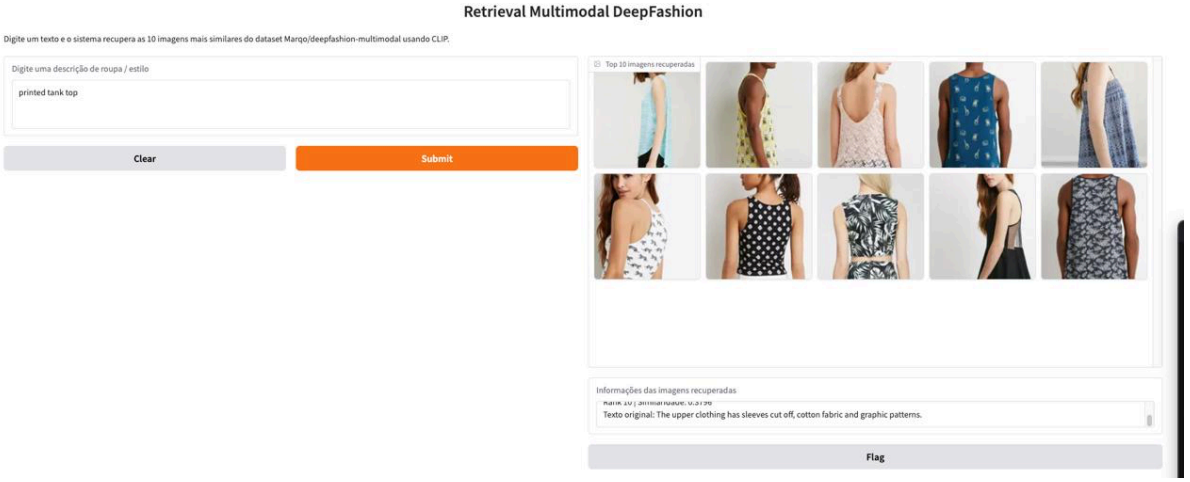
- **1. Demonstração de Métricas de Retrieval**

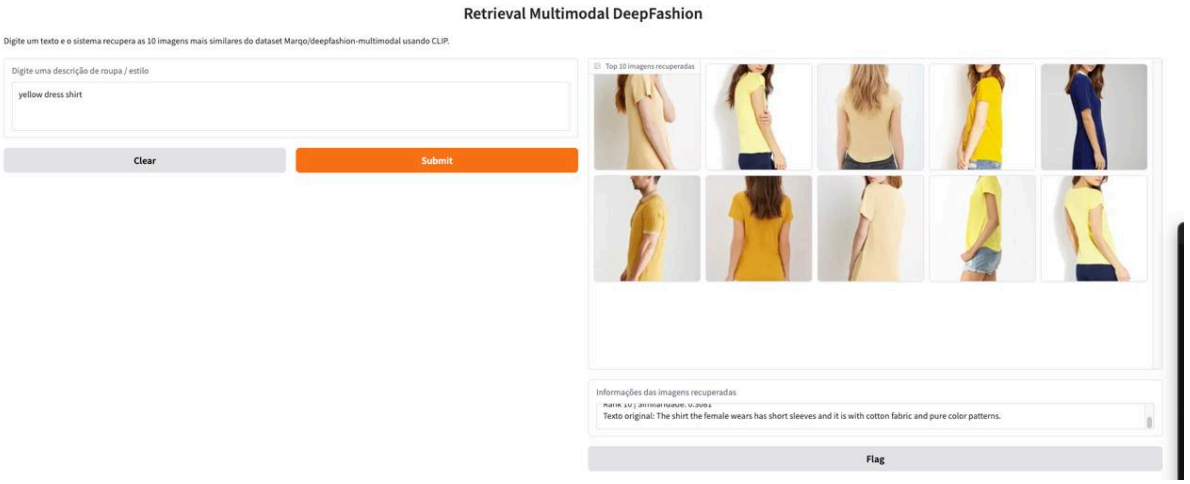


- **2. Demonstração de Retrieval**
- **2.1**

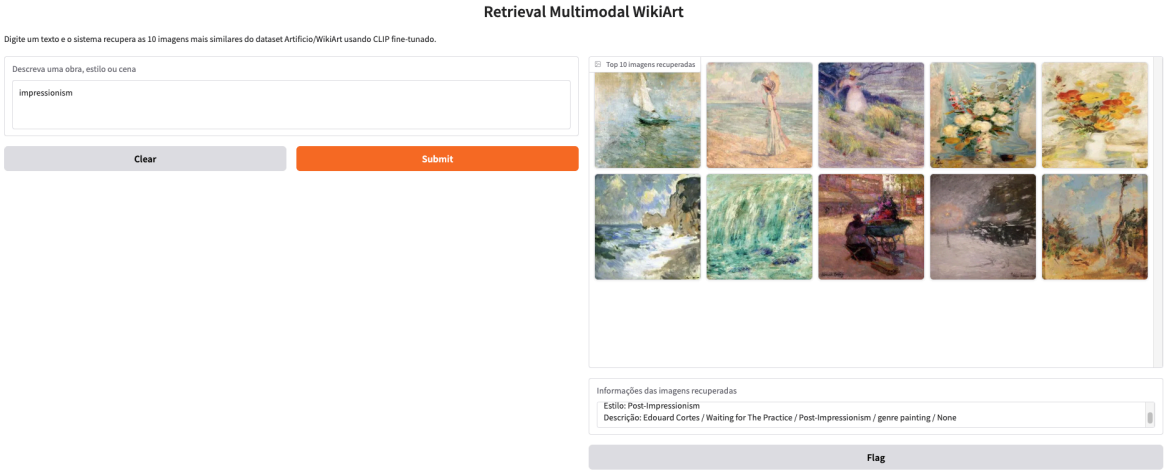


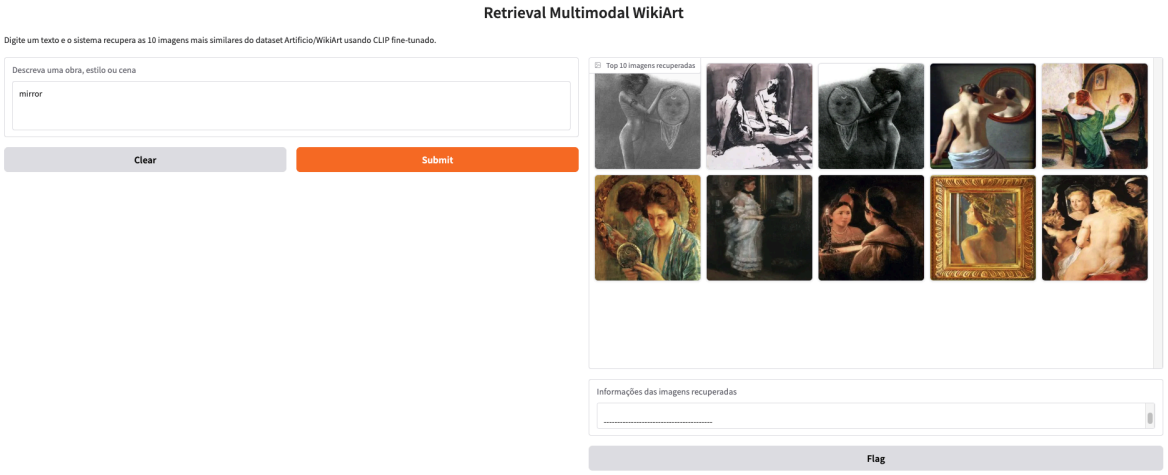

Retrieval Multimodal DeepFashion
Digite um texto e o sistema recupera as 10 imagens mais similares do dataset Marqo/deepfashion-multimodal usando CLIP.
Digite uma descrição de roupa / estilo
stylish skirt
Clear
Submit
Top 10 imagens recuperadas
Informações das imagens recuperadas
is of medium length. The skirt is with cotton fabric and graphic patterns. There is a ring on her finger.
Flag

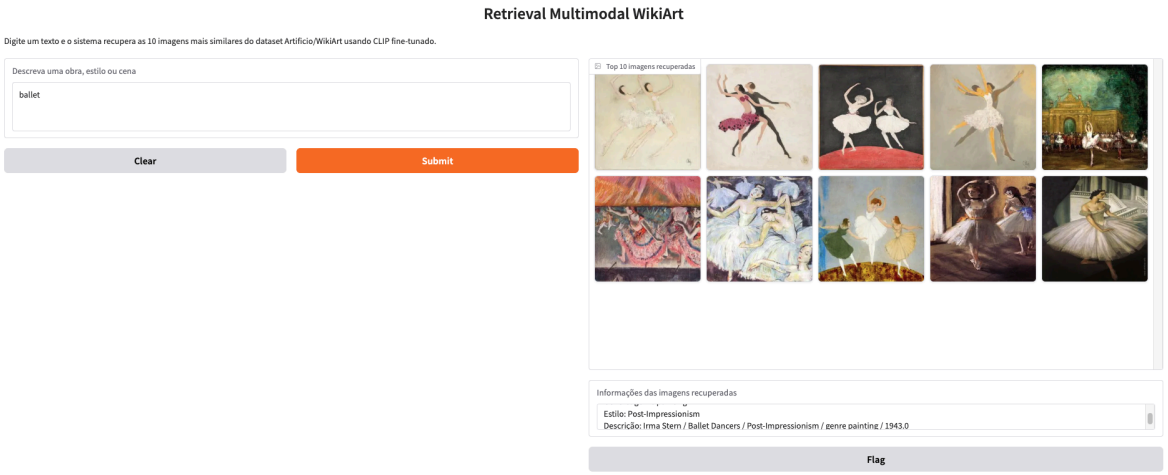

Retrieval Multimodal DeepFashion
Digite um texto e o sistema recupera as 10 imagens mais similares do dataset Marqo/deepfashion-multimodal usando CLIP.
Digite uma descrição de roupa / estilo
printed tank top
Clear
Submit
Top 10 imagens recuperadas
Informações das imagens recuperadas
Texto original: The upper clothing has sleeves cut off, cotton fabric and graphic patterns.
Flag

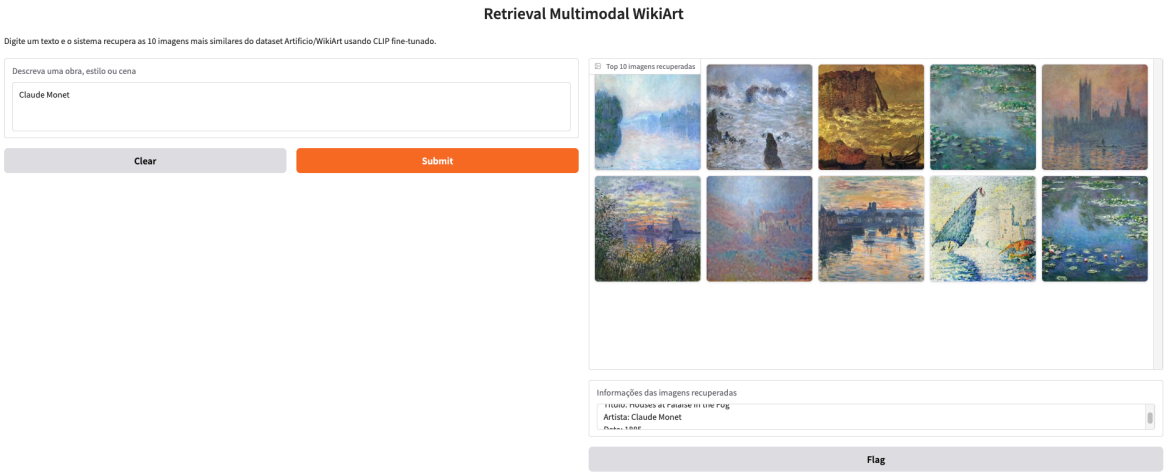

Retrieval Multimodal DeepFashion
Digite um texto e o sistema recupera as 10 imagens mais similares do dataset Marqo/deepfashion-multimodal usando CLIP.
Digite uma descrição de roupa / estilo
yellow dress shirt
Clear
Submit
Top 10 imagens recuperadas
Informações das imagens recuperadas
Texto original: The shirt the female wears has short sleeves and it is with cotton fabric and pure color patterns.
Flag

- **2.2**

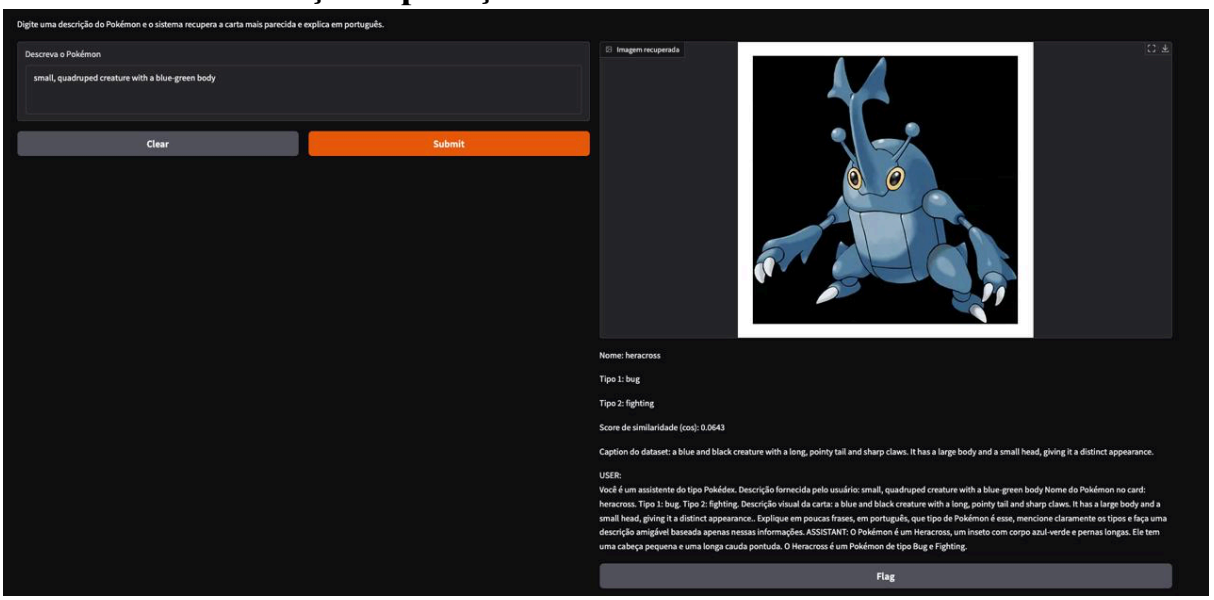

Retrieval Multimodal WikiArt
Digite um texto e o sistema recupera as 10 imagens mais similares do dataset Artificio/WikiArt usando CLIP fine-tunado.
Descreva uma obra, estilo ou cena
impressionism
Clear
Submit
Top 10 imagens recuperadas
Informações das imagens recuperadas
Estilo: Post-Impressionism
Descrição: Edouard Cortes / Waiting for The Practice / Post-Impressionism / genre painting / None
Flag

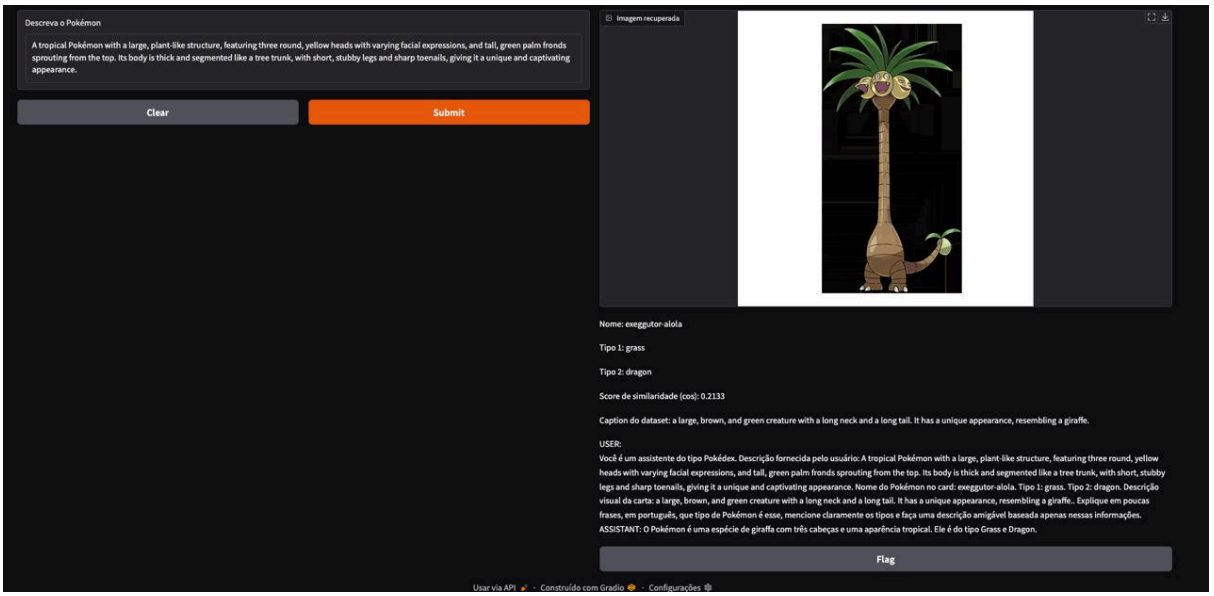

Retrieval Multimodal WikiArt
Digite um texto e o sistema recupera as 10 imagens mais similares do dataset Artificio/WikiArt usando CLIP fine-tunado.
Descreva uma obra, estilo ou cena
mirror
Clear
Submit
Top 10 imagens recuperadas
Informações das imagens recuperadas
Flag

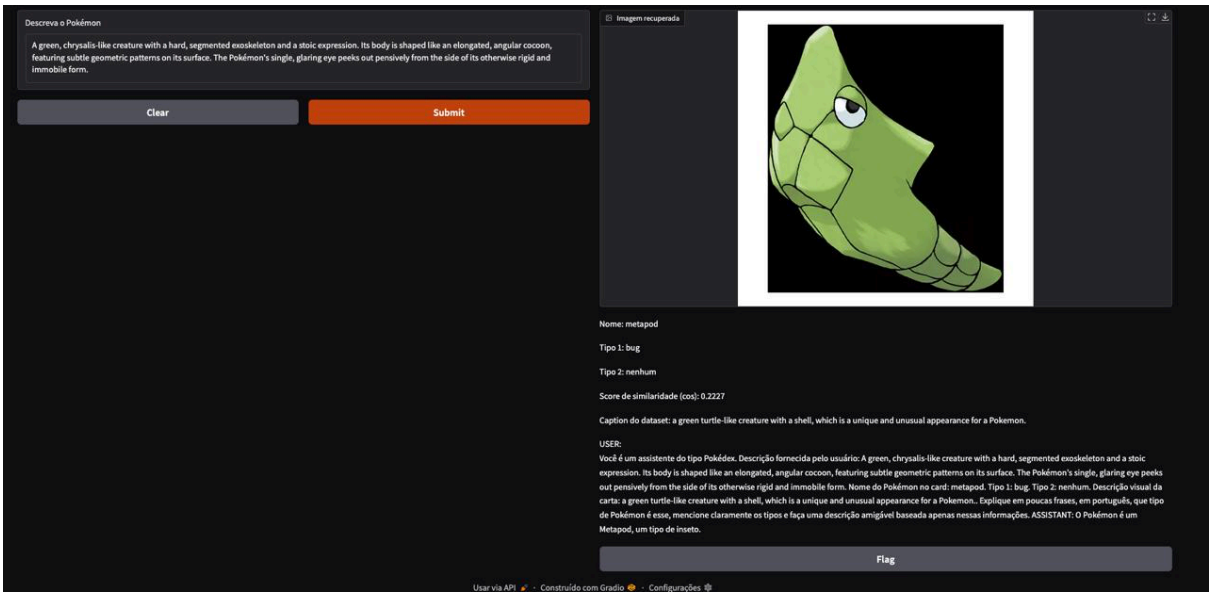

Retrieval Multimodal WikiArt
Digite um texto e o sistema recupera as 10 imagens mais similares do dataset Artificio/WikiArt usando CLIP fine-tunado.
Descreva uma obra, estilo ou cena
ballet
Clear
Submit
Top 10 imagens recuperadas
Informações das imagens recuperadas
Estilo: Post-Impressionism
Descrição: Irma Stern / Ballet Dancers / Post-Impressionism / genre painting / 1943.0
Flag

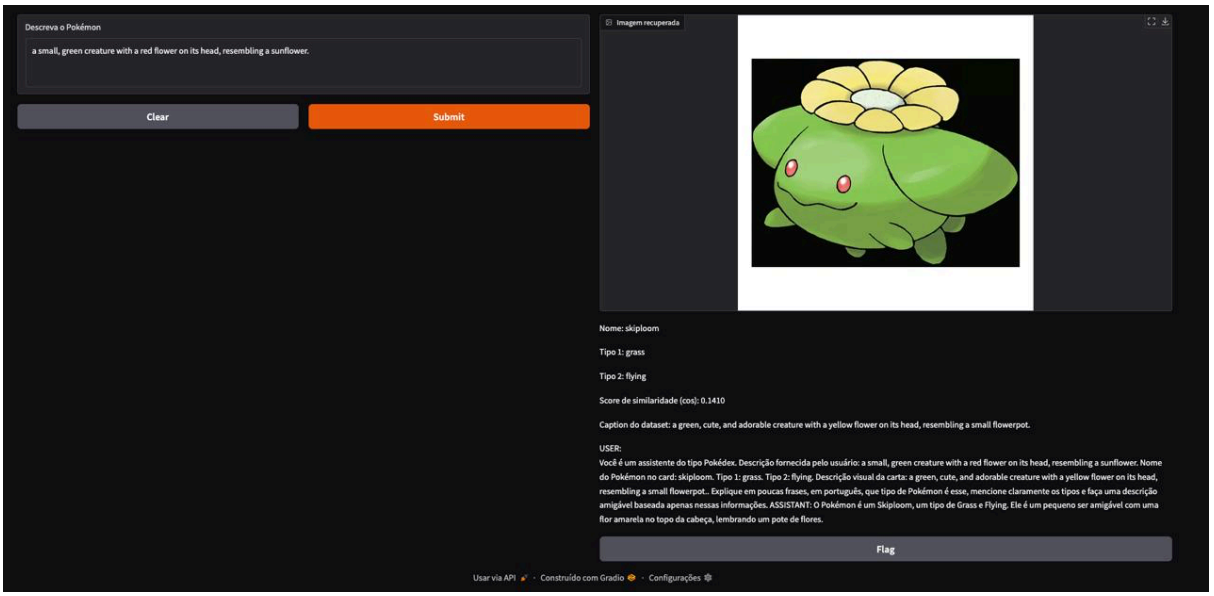


- 3. Demonstração/Aplicação de RAG




Descreva o Pokémon
A tropical Pokémon with a large, plant-like structure, featuring three round, yellow heads with varying facial expressions, and tall, green palm fronds sprouting from the top. Its body is thick and segmented like a tree trunk, with short, stubby legs and sharp toenails, giving it a unique and captivating appearance.
Clear
Submit
Imagem recuperada
Nome: exeggutor-alola
Tipo 1: grass
Tipo 2: dragon
Score de similaridade (cos): 0.2133
Caption do dataset: a large, brown, and green creature with a long neck and a long tail. It has a unique appearance, resembling a giraffe.
USER:
Você é um assistente do tipo Pokédex. Descrição fornecida pelo usuário: A tropical Pokémon with a large, plant-like structure, featuring three round, yellow heads with varying facial expressions, and tall, green palm fronds sprouting from the top. Its body is thick and segmented like a tree trunk, with short, stubby legs and sharp toenails, giving it a unique and captivating appearance. Nome do Pokémon no card: exeggutor-alola. Tipo 1: grass. Tipo 2: dragon. Descrição visual da carta: a large, brown, and green creature with a long neck and a long tail. It has a unique appearance, resembling a giraffe.. Explique em poucas frases, em português, que tipo de Pokémon é esse, mencione claramente os tipos e faça uma descrição amigável baseada apenas nessas informações.
ASSISTANT: O Pokémon é uma espécie de girafa com três cabeças e uma aparência tropical. Ele é do tipo Grass e Dragon.
Flag
Usar via API · Construído com Gradio · Configurações


Descreva o Pokémon
A green, chrysalis-like creature with a hard, segmented exoskeleton and a stoic expression. Its body is shaped like an elongated, angular cocoon, featuring subtle geometric patterns on its surface. The Pokémon's single, glaring eye peeks out pensively from the side of its otherwise rigid and immobile form.
Clear
Submit
Imagem recuperada
Nome: metapod
Tipo 1: bug
Tipo 2: nenhum
Score de similaridade (cos): 0.2227
Caption do dataset: a green turtle-like creature with a shell, which is a unique and unusual appearance for a Pokemon.
USER:
Você é um assistente do tipo Pokédex. Descrição fornecida pelo usuário: A green, chrysalis-like creature with a hard, segmented exoskeleton and a stoic expression. Its body is shaped like an elongated, angular cocoon, featuring subtle geometric patterns on its surface. The Pokémon's single, glaring eye peeks out pensively from the side of its otherwise rigid and immobile form. Nome do Pokémon no card: metapod. Tipo 1: bug. Tipo 2: nenhum. Descrição visual da carta: a green turtle-like creature with a shell, which is a unique and unusual appearance for a Pokemon.. Explique em poucas frases, em português, que tipo de Pokémon é esse, mencione claramente os tipos e faça uma descrição amigável baseada apenas nessas informações. ASSISTANT: O Pokémon é um Metapod, um tipo de inseto.
Flag
Usar via API · Construído com Gradio · Configurações



Lógica DEMONSTRAÇÃO 1

1. PADRÃO DOS TEXTOS

A lógica usa:

- total_linhas
- num_unicos_texto
- proporcao_unicos = num_unicos_texto / total_linhas

Casos:

a) total_linhas > 200 E (num_unicos_texto < 15 OU proporcao_unicos < 0.1)
→ Interpretação: poucos valores únicos de texto, muitas repetições.
→ Sugestão:

- “Parece mais um cenário de classificação.”

- Além de retrieval, vale considerar treinar um modelo de classificação com essas classes de texto.

b) Caso contrário (não entra na condição acima)
→ Interpretação: diversidade razoável de textos.
→ Sugestão:

- “Combina bem com tarefas de retrieval ou geração multimodal.”

2. DESEMPENHO (RECALL@1)

Pegamos o melhor modelo (maior recall@1) e classificamos:

- r1 < 0.10 → nível: “muito baixo”
- 0.10 ≤ r1 < 0.20 → nível: “baixo”
- 0.20 ≤ r1 < 0.35 → nível: “moderado”
- r1 ≥ 0.35 → nível: “alto”

Isso é só o rótulo. As sugestões principais vêm da combinação com o tamanho do dataset (próxima seção).

Além disso, há a comparação de famílias:

- Se tiver SigLIP e CLIP:
    - Se média R@1 SigLIP > R@1 CLIP + 0.02 → sugerimos: “SigLIP está melhor que CLIP neste dataset.”
    - Se média R@1 CLIP > R@1 SigLIP + 0.02 → sugerimos: “CLIP está melhor que SigLIP neste dataset.”
    - Diferença menor que 0.02 → sugerimos: “Desempenho médio parecido entre as famílias.”

3. FINE-TUNING (USANDO RECALL@1 + TAMANHO DO DATASET)

- pequeno: total_linhas < 300
- médio/grande: 300 ≤ total_linhas ≤ 100000
- muito grande: total_linhas > 100000

E cruzamos com o recall@1 do melhor modelo:

Caso 1: r1 < 0.15 (performance baixa)
1.1) total_linhas < 300
→ Sugestão:
- "Recall@1 baixo e dataset pequeno. Prioridade é aumentar o dataset antes de investir em fine-tuning."
- Se o padrão de texto for de classificação (seção 1), reforçamos a ideia de modelo de classificação também.

1.2) 300 ≤ total_linhas ≤ 100000
→ Sugestão:
- "Recall@1 baixo, dataset razoável. Recomenda-se fine-tuning imediato focado nesse domínio."
- Se SigLIP ou CLIP estiver melhor, a leitura natural é: fine-tuning em cima da família que foi melhor.

1.3) total_linhas > 100000
→ Sugestão:
- "Recall@1 baixo e dataset gigantesco. Recomenda-se abordagem robusta:
- pré-treinamento adicional,
- amostragem cuidadosa,
- técnicas específicas para grandes volumes (curriculum, sampling, etc.)."

Caso 2: 0.15 ≤ r1 < 0.35 (performance moderada)
→ Sugestão (independente do tamanho, mas com nuance implícita):

- "Recall@1 moderado. Um fine-tuning provavelmente trará ganhos claros."
- Se dataset for muito pequeno, o ganho pode ser limitado; se for maior, FT é bem promissor.

Caso 3: r1 ≥ 0.35 (performance alta)
→ Sugestão:

- "Recall@1 alto. Fine-tuning não é necessário, mas pode refinar o comportamento para produção."
- Foco mais em deploy/uso prático do que em mexer no modelo.

4. RESUMO EM FORMATO DIRETO
   Para qualquer dataset:
1. Checamos se o padrão de textos parece classificação:
    - Se sim: sugerimos considerar modelo de classificação.
    - Se não: sugerimos retrieval/geração como principal.
2. Medimos o melhor recall@1:
    - < 0.15 → consideramos performance baixa.
    - 0.15–0.35 → moderada.
    - ≥ 0.35 → alta.
3. Combinamos com o tamanho do dataset:
    - Pequeno (<300):
        - Baixo R@1: "colete mais dados, FT agora pode não ser ótimo."
    - Médio/Grande (300–100k):
        - Baixo R@1: "recomendado fine-tuning direto."
        - Moderado R@1: "fine-tuning deve trazer melhoria clara."
    - Muito grande (>100k):

        - Baixo R@1: “precisa estratégia robusta (pré-treinamento, sampling).”

    - Comparamos média SigLIP vs CLIP e dizemos qual família está indo melhor para o dataset do usuário, para orientar qual modelo ajustar.