Avaliação da quantização de modelos visuais de linguagem para implantação em ambiente hospitalar

Rafael Scalabrin Dosso, Diedre Santos do Carmo, Letícia Rittner
Faculdade de Engenharia Elétrica e de Computação, Universidade de Campinas (UNICAMP)
Campinas, SP, Brasil

***Resumo*—A implantação de modelos de Inteligência Artificial de ponta em hospitais é limitada por questões éticas e altos custos computacionais, especialmente para instituições menores e de países com restrições de recursos. Este estudo avalia a quantização como estratégia para viabilizar a execução local de Modelos de Visão e Linguagem (VLMs) para a interpretação de radiografias de tórax. Utilizando os modelos CheXagent, CheXagent 2 e MedGemma, aplicamos Quantização Pós-Treinamento (PTQ) de 4 bits e analisamos o trade-off entre acurácia (F1-Score), uso de memória (VRAM) e tempo de inferência. Os resultados mostram que a quantização reduziu drasticamente o uso de VRAM, com o MedGemma demonstrando menor perda de acurácia em relação ao CheXagent. O CheXagent 2, não quantizado, alcançou o maior desempenho e menor tempo de inferência. Embora o CheXagent 2 seja relativamente leve, seus requisitos de hardware ainda limitam a sua implantação e reforçam que a quantização é uma ferramenta viável para adaptar megamodelos avançados a possíveis limitações do ambiente clínico.**

***Abstract*—The deployment of state-of-the-art Artificial Intelligence models in hospitals is limited by ethical issues and high computational costs, especially for smaller institutions and in resource constrained countries. This study evaluates quantization as a strategy to enable the local execution of Vision and Language Models (VLMs) for the interpretation of chest radiographs. Using the CheXagent, CheXagent 2, and MedGemma models, we apply 4-bit Post-Training Quantization (PTQ) and analyze the trade-off between accuracy (F1-Score), memory usage (VRAM), and inference time. The results show that quantization drastically reduced VRAM usage, with MedGemma demonstrating less accuracy loss compared to CheXagent. The non-quantized CheXagent 2 achieved the highest accuracy and lowest inference time. Although CheXagent 2 is relatively lightweight, its hardware requirements still limit its deployment and reinforce that quantization is a viable tool for adapting advanced megamodels to the potential limitations of a clinical environment.**

I. Introdução

Alavancada pela grande popularização do uso de modelos baseados em inteligência artificial (IA) e aprendizado profundo nos mais diversos domínios, tem surgido uma crescente demanda por recursos computacionais necessários para a utilização desta tecnologia. Especificamente em modelos que utilizam Transformers [1], estudos como "Scaling Laws for Neural Language Models" [2] mostram que o seu desempenho está extremamente atrelado a um aumento acentuado do número de dados e parâmetros empregados. Consequentemente, à medida que os modelos modernos se desenvolvem para maior exatidão, surgem preocupações de custos operacionais relacionados às suas demandas infraestruturais.

No contexto hospitalar, especialmente em países menos desenvolvidos, é comum não haver a disponibilidade dos mesmos recursos de processamento de ambientes acadêmicos ou corporativos em que os modelos são elaborados [3]. Cotidianamente, arquiteturas como o ChatGPT [4] e o DeepSeek [5] são executadas remotamente em serviços de nuvem, o que é indesejável no ambiente hospitalar por dois principais motivos: custos recorrentes de utilização remota; e confidencialidade dos dados [6]. Esse contexto muitas vezes reflete numa preferência por execução local, que é impedida frequentemente por limitações de infraestrutura local.

Nesse sentido, uma solução condizente com este cenário é a quantização [7] [8] dos modelos, técnica que visa aplicar transformações nos pesos de uma rede neural para diminuir significativamente o seu uso de memória, sem grandes comprometimentos de precisão em tempo de inferência. Assim, viabiliza-se a utilização de modelos modernos em hardware limitado, possibilitando a execução de inferências em ambientes restritos como o hospitalar.

Este trabalho investiga a quantização como maneira de preencher a lacuna que separa os modelos acadêmicos modernos das realidades hospitalares, em virtude da disparidade de recursos computacionais e preocupações de confidencialidade. Este cenário se acentua ainda mais em contextos de países em desenvolvimento e em instituições da rede pública, como é o caso do Hospital de Clínicas de Porto Alegre (HCPA) [1], hospital para o qual esta pesquisa foi direcionada. Ainda assim, acredita-se que os resultados obtidos possam contribuir para o uso acessível de recursos de IA na saúde pública nacional e internacional.

Nessa perspectiva, foi avaliado o impacto da quantização em modelos visuais de linguagem modernos (VLMs, do inglês Vision Language Model), como o CheXagent [9] e o MedGemma [10], aplicados a radiografias de tórax (CXRs), buscando analisar a relação entre custo computacional e desempenho preditivo dos modelos após quantização.

II. Método

O HCPA demonstrou principal interesse no uso de VLMs para caracterização de Raios X. Os modelos avaliados foram o CheXagent [9], com 8B de parâmetros, o MedGemma [10] com 4B de parâmetros, e o CheXagent 2[2], uma evolução do CheXagent com 3B de parâmetros que reduziu muito seu uso computacional ao modernizar a arquitetura do LLM

[1]Página do HCPA: https://www.hcpa.edu.br/

[2]https://huggingface.co/StanfordAIMI/CheXagent-2-3b

utilizado. Os modelos foram utilizados para avaliar o impacto da quantização na tarefa de identificação de 14 achados patológicos nas radiografias, tarefa consolidada na literatura por refletir um cenário clínico relevante e por estar bem estabelecida em conjuntos de dados.

A. Conjuntos de dados

Os conjuntos de dados utilizados foram o CheXpert [11], o MIMIC-CXR [12] e o BRAX [13], que contêm no total cerca de 600 mil amostras, bem como 14 rótulos indicando presença de achados radiológicos derivados de laudos hospitalares, sendo eles: Aumento do Cardiomediastino, Cardiomegalia, Lesão Pulmonar, Opacidade Pulmonar, Edema, Consolidação, Pneumonia, Atelectasia, Pneumotórax, Derrame Pleural, Outro Pleural, Fratura, Dispositivos de Suporte e Nenhum destes achados. Os modelos foram avaliados por imagem, sobre duas principais amostragens:

- **Conjunto de Teste Preliminar:** 234 imagens do conjunto de validação do CheXpert, (que, apesar da nomenclatura oficial do conjunto pelos seus autores, elas foram utilizadas para teste e não para validação). Esta amostragem foi utilizada para experimentação inicial de definição do método e parâmetros, graças à sua menor quantidade de aquisições e anotação 100% manual.
- **Conjunto de Teste Agregado:** uma seleção aleatória de 30.229 imagens provenientes dos três conjuntos de dados citados (CheXpert, MIMIC-CXR e BRAX), todas com anotações automáticas baseadas no processamento dos laudos radiológicos originais. Utilizada para avaliações finais mais substanciais por causa do seu maior número de amostras.

B. Pipelines para extração de patologias

Para empregar o CheXagent e o MedGemma para a identificação binária dos 14 achados específicos, dois principais métodos foram desenvolvidos e aplicados:

1) **Identificação Direta (Pipeline 1):** aqui, os VLMs (CheXagent ou MedGemma) recebem, juntamente com imagem, um *prompt* pedindo para que identifiquem os achados patológicos presentes. O texto de saída é então convertido para um vetor que contém a presença ou ausência de cada patologia. A vantagem dessa abordagem é que há poucas brechas para interpretação da saída, que simplesmente citará os achados presentes, facilitando a obtenção dos achados após inferência (Fig. 1).

2) **Geração de Relatório + CheXbert (Pipeline 2):** neste pipeline, o VLM recebe a imagem com um prompt, o qual desta vez solicita um relatório médico a ele. Então, este relatório é submetido ao CheXbert [14], modelo de processamento textual que identificará a presença ou ausência dos achados no texto livre obtido. Este método visa se aproveitar da natureza dos VLMs utilizados, que têm um grande foco em geração de texto livre. Assim, eles são utilizados no seu contexto principal de uso, às custas do surgimento de um processo adicional, que pode introduzir mais uma fonte de imprecisão (Fig. 2).

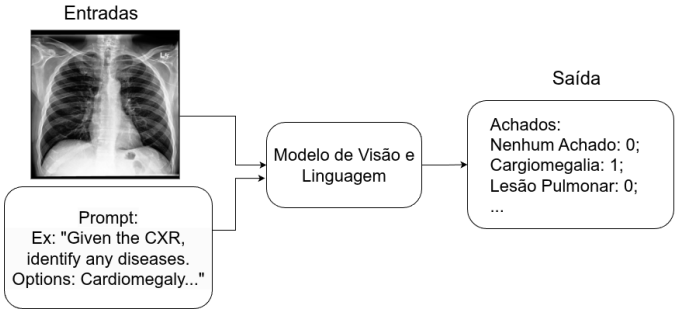

Figura 1. Pipeline 1 de obtenção de achados, onde o prompt lista as possibilidades de achados e VLM devolve, dentre os achados listados, quais foram encontrados.

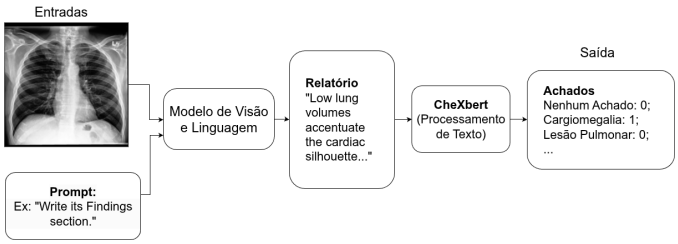

Figura 2. Método 2 de obtenção de achados, onde o prompt dá a chance para o VLM elaborar um texto livre e utiliza o modelo CheXbert para extrair os achados do texto livre

C. Quantização

A quantização é uma maneira de reduzir o uso de memória de representações numéricas. Em Aprendizado Profundo [15], ela é utilizada para fazer com que os pesos dos modelos, que utilizam tipos numéricos como *float32* e *bfloat16*, sejam representados com tipos menores como *int8* e *int4*. A etapa que mapeia cada valor de um intervalo numérico para o outro é chamada de calibração, e há diferentes métodos de fazer essa conversão.

O estudo se concentrou na chamada Quantização Pós-Treinamento (PTQ) Dinâmica, que aplica as transformações numéricas a um modelo já treinado e sem necessidade de dados adicionais ou retreinamento, embora requeira um tempo de inferência maior. Os pesos dos modelos base, originalmente no formato *bfloat16*, foram convertidos para o formato *int4*, reduzindo o uso de memória durante inferência para aproximadamente um quarto do original. A biblioteca BitsAndBytes[3] integrada ao framework HuggingFace Transformers[4] foi utilizada para esse propósito. A avaliação foi realizada em uma GPU (Graphics Processing Unit) NVIDIA A100. É importante notar que o uso de memória gráfica (VRAM)

[3]BitsAndBytes: https://huggingface.co/docs/bitsandbytes/main/en/index
[4]HuggingFace Transformers: https://huggingface.co/docs/transformers/index

durante inferência é o principal limitador da possibilidade de implantação de modelos em ambientes com recursos limitados. Quantidades comuns de memória em hardware de baixo custo variam entre 8, 12 e 16 GB, com GPUs com 32 GB ou mais custando de 10 a 100 vezes mais no Brasil. Para medir o uso máximo de VRAM, utilizamos-nos do comando nvidia-smi durante inferência, incluindo prompt e uma imagem de raios-x na entrada.

III. Resultados e Discussão

Os primeiros experimentos foram executados com o conjunto de Teste Preliminar (II-A) para identificar os comportamentos relacionados à quantização, aos diferentes pipelines (II-B) e às diferenças de um modelo para o outro. Nota-se que o CheXagent 2 não suporta quantização para 4 bits, mas segue envolvido nos experimentos representando uma alternativa comum na literatura para implantação em infraestrutura limitada: simplesmente procurar por modelos já desenvolvidos com o fim de serem leves, sem necessidade de quantização. Neste caso, os autores do CheXagent treinaram o método novamente com um LLM muito mais leve.

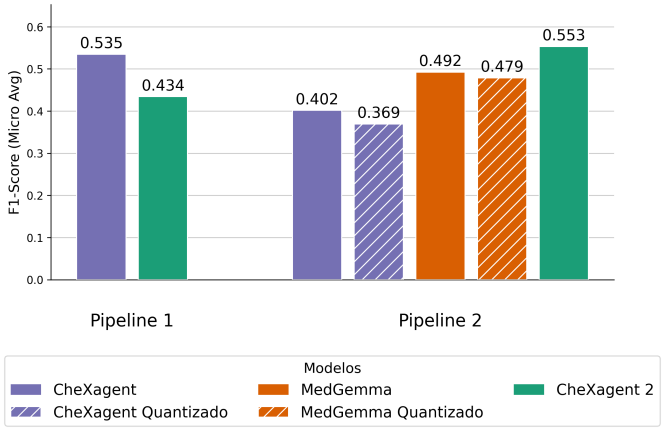

Figura 3. Gráfico de F1-score (Micro Avg) para os experimentos iniciais no **Conjunto de Teste Preliminar**, mostrando os desempenhos dos modelos com ambos os pipelines e na versão quantizada e padrão.

A figura 3 resume os experimentos no conjunto de dados de Teste Preliminar II-A. Nota-se a performance superior do CheXagent 2 pelo **Pipeline 2**, que atingiu um F1-score de 0,553. No entanto, o CheXagent padrão, pela Identificação Direta, alcançou resultado semelhante, mostrando que o desempenho de cada método de uso pode variar com o modelo, especialmente em uma amostragem pequena como esta. Similarmente, a quantização para 4 bits causou impactos de diferentes magnitudes no desempenho do CheXagent e do MedGemma, mas revelou-se uma abordagem promissora, visto que os modelos mantiveram suas performances em nível semelhante. Após esta experimentação, o Pipeline 2 foi escolhido para experimentação com o conjunto de dados principal.

Na figura 4, que resume os resultados dos modelos para o **Pipeline 2** no conjunto de Teste Agregado, vê-se novamente

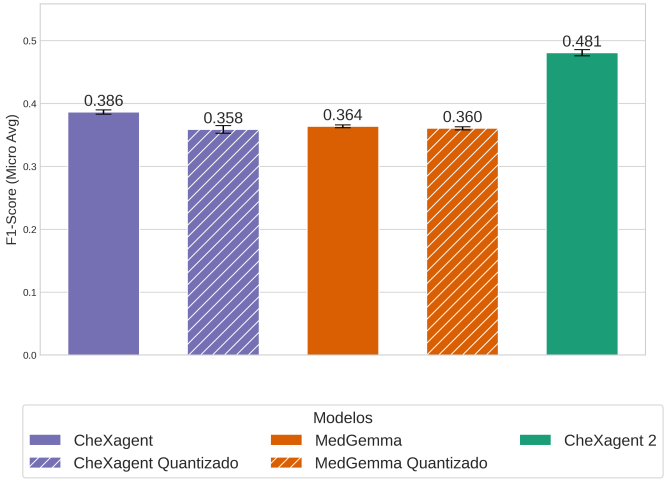

Figura 4. Gráfico de F1-score (Micro Avg) e Desvio Padrão em 5 Folds do **Conjunto de Teste Agregado** e **Pipeline 2**

a superioridade em acurácia do CheXagent 2, que atingiu F1-score de 0,481. As mudanças de arquitetura implementadas no CheXagent 2 trouxeram resultados extremamente expressivos, uma vez que aumentaram o desempenho significativamente e reduziram o número de parâmetros de 8 bilhões para 3 bilhões. Claro que a otimização de um modelo existente com ajuste-fino e arquitetura mais eficiente impõe custo alto de desenvolvimento e treino e nem sempre é uma solução viável.

A quantização mostrou-se uma opção sólida nos demais modelos, com impactos diferentes no desempenho. Assim como observou-se no **Conjunto de Teste Preliminar**, vê-se que o MedGemma apresentou uma variação de precisão surpreendentemente pequena, menor do que a do CheXagent, o qual chegou a perder aproximadamente 3 pontos percentuais de acurácia.

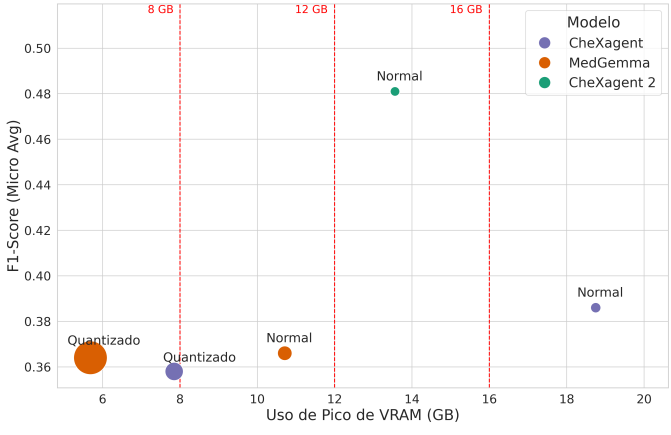

Figura 5. Gráfico de pico do uso de memória (VRAM) por desempenho (F1-micro) e tempo médio de inferência (tamanho dos pontos) dos modelos avaliados no **Conjunto de Teste Agregado** com o **Pipeline 2**. As linhas verticais em 8, 12 e 16 GB mostram as capacidades de VRAM de GPUs de baixo custo.

É crucial avaliar, além dos desempenhos brutos de cada modelo, os custos computacionais de cada um deles, fatores determinantes para a viabilização de uma implantação local em ambiente hospitalar. A figura 5 mostra os *trade-offs* entre a performance, uso de memória (VRAM) e tempo de inferência, com destaque para as linhas verticais indicando quantidades de memória comumente disponíveis em GPUs de baixo custo. Este estudo revela um achado importante: a quantização de 4 bits traz uma penalidade considerável no tempo de execução. A diferença observada foi uma inferência cerca 3 vezes mais lenta para o CheXagent e aproximadamente 6 vezes mais lenta para o MedGemma, que já tem tempo médio de inferência elevado. Acredita-se que a causa deste aumento é que, principalmente com a Quantização Pós-Treinamento Dinâmica, há um grande *overhead* de operações sendo realizadas durante a execução para converter e desconverter os pesos para algumas operações.

Além disso, nota-se que o CheXagent 2, com uma performance convincentemente superior aos demais modelos, estabeleceu-se como uma solução de excelente custo-benefício computacional, uma vez que tem uso de memória moderado e tempo de execução baixo. Nota-se a necessidade de uma GPU de 16GB de VRAM para execução do CheXagent 2. Em casos onde somente 8 GB ou 12 GB está disponível, o MedGemma quantizado aparece como uma boa alternativa, visto que tem o menor uso de VRAM enquanto mantém um desempenho competitivo e alinhado ao de sua versão base.

No contexto desta pesquisa, onde a versão 2 modernizada e mais leve do CheXagent estava disponível para uso, esta segunda versão se mostra superior, porém também demostramos que o impacto da quantização nesses tipos de modelo na performance e velocidade de interpretação de exames de raios-x é pequena em comparação com os benefícios de uso de memória e subsequente viabilização de sua execução em infraestrutura limitada.

IV. Conclusão

O estudo avaliou o impacto da Quantização Pós-Treinamento para 4 bits em Modelos de Visão e Linguagem modernos, trazendo um panorama da relação custo-benefício em termos computacionais. A quantização teve um impacto pequeno na acurácia de classificação, trazendo benefícios significativos na viabilização da implementação dos modelos analisados em hardware limitado. Além disso, o MedGemma revelou-se uma boa opção para a aplicação da quantização em ambientes extremamente limitados, não apresentando grandes perdas de desempenho e oferencendo o menor uso de memória dentre as opções analisadas.

Dentre os modelos analisados, o CheXagent 2, não quantizado, mostrou-se uma opção sólida, com o melhor F1-Score e um tempo de inferência baixo. Porém, inviável de ser utilizado com GPUs de baixo custo com VRAM abaixo de 16GB. A utilização de modelos mais recentes com arquiteturas mais otimizadas pode trazer grandes benefícios, e permanece sempre como uma alternativa à quantização de modelos maiores. Por fim, conclui-se que a quantização de modelos grandes mostra-se como uma opção viável para implantação destes modelos em ambientes com hardwares limitados e preocupações éticas, como em hospitais e clínicas.

Agradecimentos

Os autores agradecem à NeuralMind pelo financiamento do projeto. Ferramentas de IA como o Gemini e o Deepseek foram utilizadas para consulta e revisão de determinados trechos do artigo.

Referências

[1] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in Transformer," *Advances in Neural Information Processing Systems*, vol. 34, pp. 15 908–15 919, 2021.

[2] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[3] H. Neumeier, *The relation of hospital choices in IT infrastructure spending and deployment with HIT/EHR strategies and operational efficiency*. The University of Alabama at Birmingham, 2013.

[4] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020.

[5] A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan *et al.*, "DeepSeek-V3 technical report," *arXiv preprint arXiv:2412.19437*, 2024.

[6] Governo do Brasil, "Lei Geral de Proteção de Dados Pessoais (LGPD) - Lei nº 13.709, de 14 de agosto de 2018," https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm, 2018, acesso em: abr. 2025.

[7] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2325–2383, 1998.

[8] H. Wu, P. Judd, X. Zhang, M. Isaev, and P. Micikevicius, "Integer quantization for deep learning inference: Principles and empirical evaluation," *arXiv preprint arXiv:2004.09602*, 2020.

[9] Z. Chen, M. Varma, J.-B. Delbrouck, M. Paschali, L. Blankemeier, D. Van Veen, J. M. J. Valanarasu, A. Youssef, J. P. Cohen, E. P. Reis *et al.*, "CheXagent: Towards a foundation model for chest x-ray interpretation," *arXiv preprint arXiv:2401.12208*, 2024.

[10] Google, "MedGemma hugging face," https://huggingface.co/collections/google/medgemma-release-680aade845f90bec6a3f60c4, 2025, accessed: 2025-06-20.

[11] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. Ball, K. Shpanskaya *et al.*, "CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 590–597.

[12] A. E. Johnson, T. J. Pollard, N. R. Greenbaum, M. P. Lungren, C.-y. Deng, Y. Peng, Z. Lu, R. G. Mark, S. J. Berkowitz, and S. Horng, "MIMIC-CXR-JPG, a large publicly available database of labeled chest radiographs," *arXiv preprint arXiv:1901.07042*, 2019.

[13] E. P. Reis, J. P. De Paiva, M. C. Da Silva, G. A. Ribeiro, V. F. Paiva, L. Bulgarelli, H. M. Lee, P. V. Santos, V. M. Brito, L. T. Amaral *et al.*, "BRAX, Brazilian labeled chest x-ray dataset," *Scientific Data*, vol. 9, no. 1, p. 487, 2022.

[14] A. Smit, S. Jain, P. Rajpurkar, A. Pareek, A. Y. Ng, and M. P. Lungren, "CheXbert: Combining automatic labelers and expert annotations for accurate radiology report labeling using BERT," *arXiv preprint arXiv:2004.09167*, 2020.

[15] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.