

Enhancing Medical Image Report Generation through Standard Language Models: Leveraging the Power of LLMs in Healthcare

Giorgio Leonardi, Luigi Portinale* and Andrea Santomauro

Computer Science Institute, DISIT, Università del Piemonte Orientale, Alessandria (Italy)

Abstract

In recent years, Artificial Intelligence has witnessed a deep transformation, primarily driven by advancements in deep learning architectures. Among these, the Transformer architecture has emerged as a pivotal milestone, revolutionizing natural language processing and several other tasks and domains. The Transformer's ability to capture contextual dependencies across sequences, paired with its parallelizable design, made it exceptionally versatile. This plays a fundamental role in the healthcare field, where the ability to integrate and process data from various modalities, such as medical images, clinical notes and patient records, is of paramount importance in order to enable AI models to provide more informed answers. This complexity raises the demand for models that can integrate information from multiple modalities, such as text, images and audio such as multimodal transformers, which are sophisticated architectures able to process and fuse information across different modalities. Furthermore, an important goal to be achieved in the healthcare domain is to focus on pre-trained models, given the scarcity of large datasets in this field, and the need to minimise the computational resources, since healthcare organizations are not equipped with high-performance computation devices. This paper presents a methodology for harnessing pre-trained large language models based on the transformer architecture, in order to facilitate the integration of different data sources, with a specific focus on the fusion of radiological images and textual reports. The ensuing approach involves the fine-tuning of pre-existing textual models, enabling their seamless extension into diverse domains.

Keywords

Multimodal machine learning, Large language models, Automated radiology report generation

1. Introduction

In recent years, the field of artificial intelligence has witnessed a profound transformation, primarily driven by advancements in deep learning architectures. Among these, the Transformer architecture has emerged as a pivotal milestone, revolutionizing natural language processing and numerous other domains. A Transformer is extremely able to capture contextual dependencies across sequences; this feature, together with its parallelizable design, has rendered this deep

HC@AIxIA 2023: 2nd AIxIA Workshop on Artificial Intelligence For Healthcare

*Corresponding author.

† Andrea Santomauro is a PhD student enrolled in the National PhD in Artificial Intelligence for Health and Life Sciences, XXXVII cycle (Università Campus Bio-Medico di Roma)

‡ Andrea Santomauro's PhD research is co-financed by ARLANIS REPLY.

✉ giorgio.leonardi@uniupo.it (G. Leonardi); luigi.portinale@uniupo.it (L. Portinale); andrea.santomauro@uniupo.it (A. Santomauro)



© 2023 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

learning architecture exceptionally versatile. However, as the complexity of tasks in AI continues to evolve, so too does the demand for models having the ability to integrate information from multiple modalities, such as text, images, and audio. Multimodal transformers, are a potential answer to the above issues; they are of great significance in the healthcare field due to their ability to integrate and process data from various modalities, such as medical images, clinical notes, and patient records. This multimodal approach holds great potential for enhancing diagnosis, treatment, and healthcare research in general. However, in this context, several challenges have to be addressed:

- **Computational Complexity:** the employment of large and complex architectures requires substantial computational demands, and the need of significant computing resources. These computational requirements are often cost-prohibitive, hindering the widespread adoption of such models.
- **Data Scarcity:** the availability of sufficient data for training is often limited, making customized transformer training a non-trivial effort. The lack of data can lead to overfitting when attempting to train complex Transformer models, as the available data may not be sufficient to generalize efficiently.
- **Lack of Technical Transparency:** a noteworthy concern arises from the paucity of comprehensive technical public and open specifications. Many pioneering works in the literature refrain from publicly disclosing complete architectural details. Instead, they merely provide cursory insights into the overall structure while withholding finer-grained specifics. This opacity complicates efforts to replicate and build upon prior research, hampering the scientific community's ability to advance the field with precision.[1][2]

This paper introduces a robust methodology that stands out by emphasizing the deliberate reuse of pre-trained large language models, setting it apart from ad-hoc approaches in other methodologies. Our approach is dedicated to the integration of diverse data sources, with a special emphasis on merging radiological images and textual reports. In particular, we focus on the definition and experimentation of a multimodal architecture to automatically generate natural language radiological reports on the basis of radiological (X-ray) images. A key peculiarity lies in a principled fine-tuning of pre-existing textual models, ensuring their effective extension into a specific and particular healthcare domain.

2. Proposed architecture

The base model we used is GPT-2, that is a well-known decoder-only transformer. A decoder-only transformer is a neural network architecture that is derived from the original model, introduced in [3]. The Transformer architecture has become a fundamental building block in natural language processing (NLP) and has been adapted for various sequence-to-sequence tasks, including machine translation, text generation, and many more. Among the various available transformer architectures, we have excluded all encoder-only models, as the ultimate objective is text generation, necessitating decoder-only architectures. The selection of GPT-2 is attributed to its ease of use at the time of the investigation, while the most current architectures were not open source.

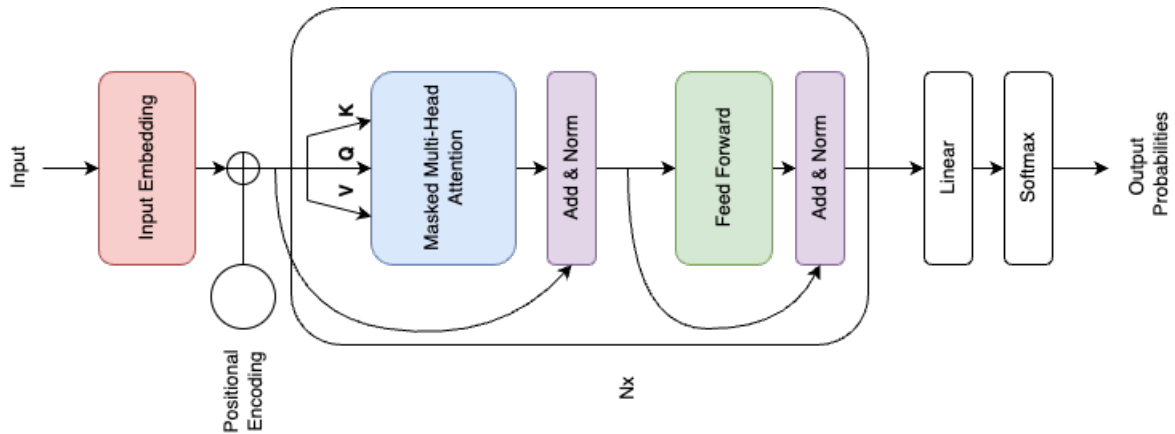


Figure 1: Decoder-only transformer (Nx means the decoder blocks are stacked Nx times)

A traditional Transformer model contains both an encoder and a decoder structure. The encoder processes the input sequence (e.g., a source sentence in machine translation), while the decoder generates the output sequence (e.g., a target sentence in machine translation). On the other hand, a decoder-only transformer is composed by:

- **Input Embedding:** like the original Transformer, the decoder-only Transformer starts by embedding the input tokens (e.g., words or subwords) into continuous vector representations.
- **Positional Encoding:** to provide information about the position of each token in the sequence, positional encodings (modeled through sine and cosine functions of different frequencies) are added to the input embeddings.
- **Multi-Head Self-Attention Mechanism:** the core component of the decoder is the multi-head self-attention mechanism, which allows the model to attend to different parts of the input sequence and capture contextual information. The decoder attends to the previously generated tokens in an autoregressive manner, that is it generates one token at a time and uses the generated tokens as context for generating subsequent tokens.
- **Masked Self-Attention:** in the decoder, a mask is applied to the self-attention mechanism to ensure that tokens cannot attend to future tokens. This is important for autoregressive generation, because each token should only depend on the tokens generated before it.
- **Multi-Head Attention Layers:** the multi-head self-attention mechanism is usually followed by feedforward neural networks for each token position. These feedforward networks can have multiple layers.
- **Layer Normalization and Residual Connections:** layer normalization and residual connections are applied after each sub-layer (e.g., self-attention and feedforward layers) to stabilize training and facilitate the flow of gradients.
- **Output Layer:** the output of the decoder-only Transformer is typically projected to the target vocabulary size through a linear layer followed by a softmax activation function.

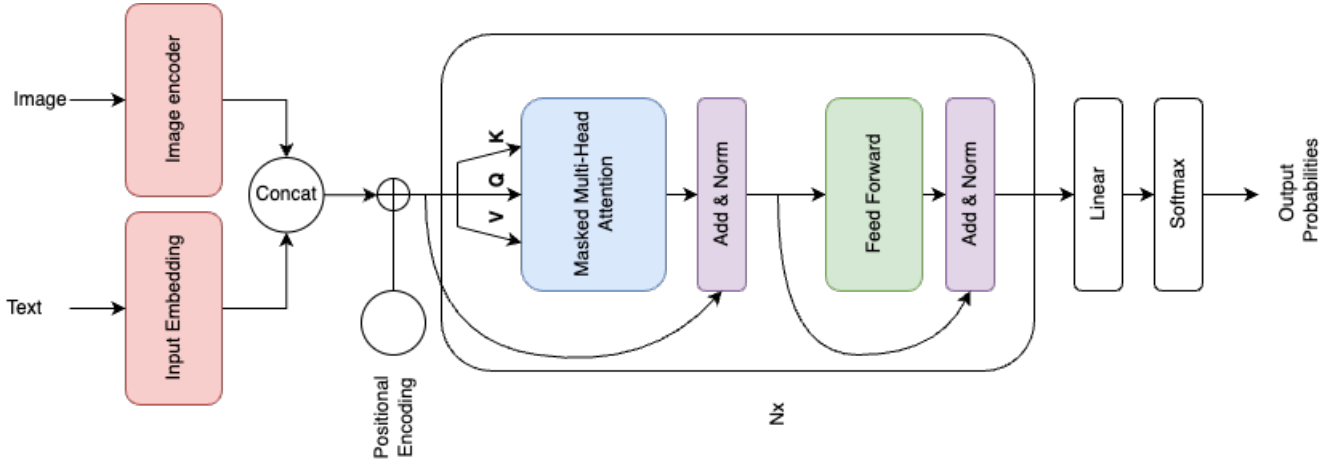


Figure 2: Multi-modal decoder-only transformer

This allows the model to generate probability distributions over the possible next tokens in the sequence.

The decoder is trained using a sequence-to-sequence task with teacher forcing, where the ground truth target sequence is used as input during training to predict the next token in the sequence. The loss function used in this particular setting is the Cross Entropy Loss; given a corpus of tokens $U = \{u_1, \dots, u_n\}$, we use a standard language modeling objective to maximize the following likelihood:

$$L(U) = \sum_i \log P(u_i | u_{i-k}, \dots, u_{i-1}; \Theta) \quad (1)$$

where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters Θ . During inference (generation), the decoder generates tokens one by one, using its own previously generated tokens as context. Greedy decoding or beam search can be used to select the next token. Decoder-only Transformers are commonly used text generation tasks and this was the reason behind the choice of this architecture. Figure 1 shows the classic decoder-only transformer architecture.

In the context of the multi-modal framework we are interested in, the aforementioned architectural configuration proves to be insufficient. Indeed, our objective is to employ a generative model in such a way that, when provided with an image as contextual input, produce a textual report, as in the task of image captioning. Consequently, an extension of the model is needed, in order to add the capability to effectively process visual information. Figure 2 depicts the model adaptation for the multi-modal setting. The modification consists in the addition of an image encoder, leaving all other components of the transformer unchanged. The input embedding, in this architecture, is composed by two steps:

- **Image embedding:** depending of which type of encoder we are using it can involve different steps. The goal is to transform the input image in a 1-d vector of size W , where W is the size of the context.

- **Text embedding:** as before.

The two different embeddings (image and text) are then concatenated to obtain a single vector V provided as input to the decoder-only Transformer. Instead of using a corpus of tokens U , we consider a set of pairs $C = \{(i, r_1), \dots, (i, r_n)\}$ where i are the images and r are the associated textual descriptions. The loss function is again the cross entropy loss, having images as fixed context, trying to predict the next token based on the previous context:

$$L(C_i) = \sum_j \log P(r_j | i, r_0..r_{j-1}; \Theta) \quad (2)$$

Note that we can use pre-trained models for both the entire transformer and the images encoder.

3. Related works

Large language models have provided significant advancements in the training of specialized models tailored for medical applications. Specific models that have emerged in this domain include BioBERT[4], ClinicalBERT[5], PubMedBERT[6], BioGPT[7], and Med-PaLM[8]. Notably, a recent addition to this landscape is the Med-Flamingo model[2][9][10], which has demonstrated remarkable performance.

These aforementioned models share three common characteristics:

- they are ad hoc models, trained from scratch.
- they necessitate a substantial volume of data for training.
- they demand a considerable amount of computational power for training and inference.

In contrast, our proposal offers an alternative approach by using pre-trained models. This approach enables fine-tuning of the model on a smaller dataset and reduces the computational resources required. Particularly, our model is trained using a single RTX6000 GPU, highlighting its efficiency in comparison to the resource-intensive nature of the aforementioned models.

4. Experiments and results

For our experiments we use a public available dataset called MIMIC CXR[11]. The MIMIC-CXR (Medical Information Mart for Intensive Care - Chest X-Ray) dataset is a large and widely used dataset in the field of medical imaging and healthcare research. It consists of chest X-ray images and associated clinical metadata, including textual reports. The dataset is composed by more than 300,000 X-ray images but it's strongly unbalanced; indeed, about 33% of the clinical studies represent normal chest X-Rays (i.e. no acute cardiopulmonary diseases are noted). In the remaining 67% there is a further imbalance between the different pathologies, with very frequent pathologies such as cardiomegaly and pulmonary edema and very rare clinical situations such as rib fractures (see Table 1). This type of imbalance is quite common in clinical datasets and can lead to low model performance.

Clinical pathology	Frequency (%)
Atelectasis	20.10%
Cardiomegaly	19.70%
Consolidation	4.75%
Edema	11.85%
Enlarged Cariomediastinum	3.15%
Fracture	1.92%
Lung Lesion	2.75%
Lung Opacity	22.61%
No Finding	33.11%
Pleural Effusion	23.85%
Pneumonia	7.26%
Pneumotorax	4.54%

Table 1
Pathology frequency in reports

We also pre-processed the MIMIC-CXR dataset with the tools CXR-RePaiR [12] and CXR-ReDonE [13]. Regarding CXR-Repair, we used the data preprocessing component to extract salient information from textual reports. Specifically, within the MIMIC-CXR dataset, the reports are organized in a way similar to complete Electronic Health Records (EHRs), and the tool made easier the extraction of the findings section related to radiological images.

Furthermore, we employed the CXR-ReDonE tool to systematically exclude any references to prior, unspecified reports. We removed these references because it was not possible the linkage of the current examination with the one referred to, due to the anonymization of the reports. Consequently, the removal of these comparative segments is crucial to mitigate the potential occurrence of erroneous associations or interpretations. We performed a rebalancing of the dataset applying a downsampling, specifically for the normal X-Ray images, obtaining about 30,000 paired data. The transformer we use is a pre-trained version of GPT-2[14], from Hugging Face (huggingface.co). As discussed in Section 2, the main difference between a standard decoder-only transformer and our architecture is the image encoder. We tested two different architectures as image encoder:

- CheXNet
- ViT input embedder

4.1. CheXNet

CheXNet is a deep neural network architecture designed for the detection of thoracic diseases, particularly chest X-ray interpretation. It was introduced in the paper [15], and it aims to assist medical professionals in diagnosing common thoracic diseases, with a focus on pneumonia detection. Figure 3 shows the CheXNet architecture.

We used the last convolutional layer as image embedding, that consist in an matrix with shape $32 \times 7 \times 7$, where 32 is the convolution depth. We then apply a flattening obtaining a matrix of 32×49 . GPT's embedding size is 768 (i.e. each "token" must have this dimensionality); for this

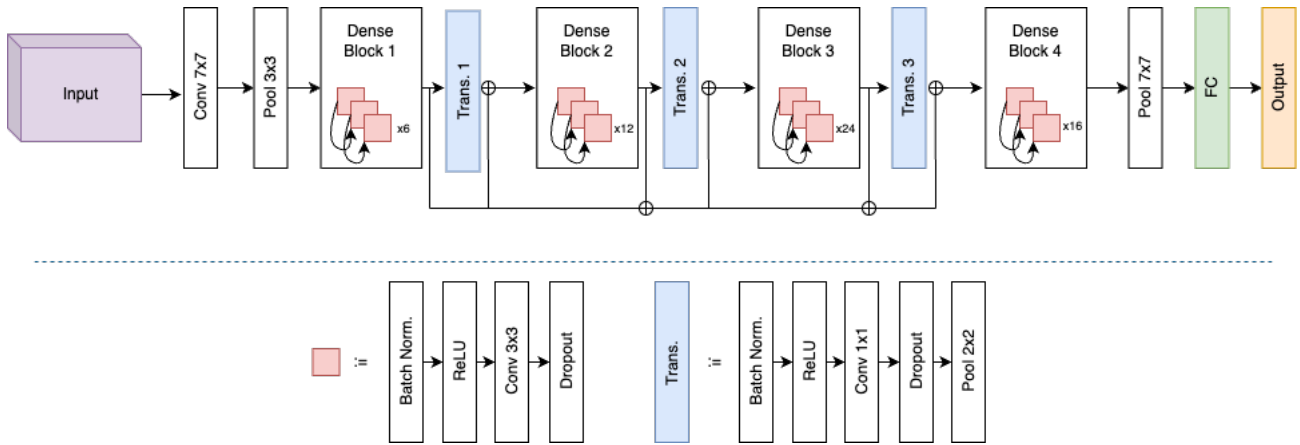


Figure 3: ChexNet Architecture: a deep convolutional neural network (CNN) architecture designed for the automated detection of diseases in chest X-ray images

reason we project the 32x49 matrix in a new matrix 32x768, having a fixed context of 32 image tokens.

4.2. ViT input embedder

The Visual Transformer[16][17], often referred to as the ViT (Vision Transformer), is a neural network architecture designed for computer vision tasks. Unlike traditional convolutional neural networks (CNNs) that process images using convolutional layers, the Visual Transformer preprocesses input images in the following way:

- Patch Extraction: the input image is divided into a grid of non-overlapping patches. Each patch is typically a small square region of the image. For example, if you have an image of size 224x224 pixels and use a patch size of 16x16, you would have 196 patches (14x14 grid).
- Flattening and Linear Projection: each patch is then flattened into a one-dimensional vector. This means that the spatial information within each patch is encoded into a linear sequence of values. These patch embeddings now serve as the input tokens to the transformer model, in this case with a dimensionality of 768.

4.3. Results

In order to evaluate our results we used a metric called **BERTScore**. BERTScore [18] is an evaluation metric for assessing the quality of machine-generated text, such as machine translation, text summarization, and more. It is designed to address some limitations of traditional evaluation metrics like BLEU and ROUGE, which often do not correlate well with human judgment of text quality. Here's a brief overview of how BERTScore works (see Figure 4):

- Pretrained BERT Model: BERTScore utilizes a pretrained BERT model [20], in order to capture contextual information from text.

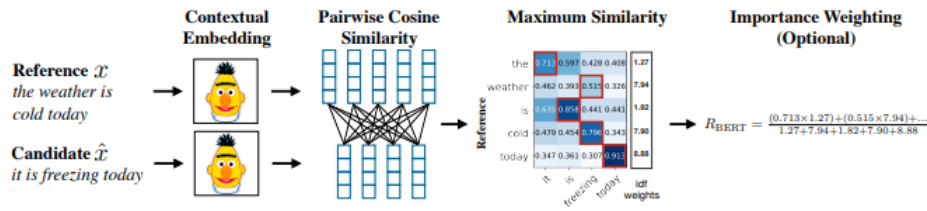


Figure 4: BERTScore’s architecture from [19]

- **Sentence Embeddings:** BERTScore tokenizes the reference and generated sentences into subword units and feeds them through the BERT model to obtain contextual embeddings for each token.
- **Cosine Similarity:** BERTScore calculates the cosine similarity between the embeddings of the reference and generated sentences. Cosine similarity measures the similarity in direction between two vectors and ranges from -1 (completely dissimilar) to 1 (identical). In this case, higher similarity scores indicate better quality.
- **Token-Level Scoring:** BERTScore computes the cosine similarity for each token in the reference and generated sentences and then computes the geometric mean of these token-level scores. This geometric mean is taken to account for the order and structure of words in the sentences.
- **Aggregation:** BERTScore can be aggregated at the sentence level to obtain a single score for the entire sentence. This is typically done by averaging the token-level scores.

One notable advantage of BERTScore is its ability to capture semantic and contextual information, which makes it more aligned with human judgment. Additionally, it doesn’t require exact matches and is more robust to variations in word choice and word order; it has become a popular metric for evaluating the quality of text generation tasks and has been used in various natural language processing applications. It provides a more reliable and interpretable assessment of generated text compared to traditional metrics. In addition, classical NLP metrics, such as BLEU score, are not effective in the context of our application, since the final goal is to obtain generated reports semantically close to the real reports. In Table 2 we can see, some examples of real and predicted reports; for example, by considering the case in (1,3), if we evaluate the predicted report with respect to the real one using BLEU score, we get a pretty low score since there are no equal n-grams; however the two reports are indicating the same clinical situation. The same argument can be made for case in (1,4).

Table 3 shows the BERTScore for the different architectures using different generation decoding type. The best results are obtained using ViT encoder and GPT-2, generating the text using a beam search decoder. All the results are obtained using AdamW as optimizer with learning rate equals to $5e-4$, epsilon equals to $1e-8$, weight decay equals to $1e-2$ and using early stopping on validation loss in order to avoid overfitting.

Examples reported in Table 2 are obtained using this combination.





X-ray	caption	X-ray	caption
	<p>Real:Right upper lobe pneumonia or mass. Recommend chest ct with intravenous contrast for further assessment. Dr. XXX communicated the above results to dr. XXX at 8 : 55 am on XXX by telephone.</p> <p>Predicted: Right middle lobe pneumonia. Recommend followup to resolution. Recommendation (s) : followup to resolution. Notification : the findings were discussed by dr. XXX (1,1)</p>		<p>Real: Mild bibasilar atelectasis. No signs of free air below the right hemidiaphragm.</p> <p>Predicted: Mild bibasilar atelectasis. No evidence of free air beneath the diaphragms. No free air under the diaphragms. (1,2)</p>
	<p>Real: Ap and lateral chest : normal heart, lungs, hila, mediastinum and pleural surfaces.</p> <p>Predicted: Cardiac and mediastinal contours stable. Lungs appear well inflated without evidence of focal airspace consolidation, pleural effusions, pulmonary edema or pneumothorax (1,3)</p>		<p>Real: 1. Upper lobe collapse and some lower lobe atelectasis around a large obstructing left hilar mass. 2. Probable small bilateral pleural effusions.</p> <p>Predicted: There is a left pleural effusion. There is no pneumothorax. There is atelectasis at the left lung base. (1,4)</p>

Table 2
Examples of Chest X-ray with real and predicted reports

Model	Decoding type	Precision	Recall	F1-score
ViT + GPT-2	Greedy search	0.75	0.73	0.74
ViT + GPT-2	Beam search	0.79	0.76	0.78
CheXNet + GPT-2	Greedy search	0.69	0.69	0.69
CheXNet + GPT-2	Beam search	0.70	0.68	0.69

Table 3
Results obtaining using different models and decoding types, using 80-20 dataset split

5. Discussions and future works

In this paper, we presented a multimodal architecture to automatically generate natural language reports on the basis of radiological images. In particular, the embeddings of chest RX images and their textual reports form the multimodal base to train a transformer-based model, able to generate new reports describing the findings detected in RX images given as query.

An important goal of this work was to focus on pre-trained models to challenge two main problems affecting the healthcare domain: (1) the problem of scarcity of data, which is common in this field, and (2) to minimise the computational resources required by our system, since the power of the machines usually available in the healthcare organizations does not allow for heavy computation. For the sake of comparison, our models are trained on a single NVIDIA RTX6000 GPU, while ad-hoc complex models, instead, are trained on multiple GPUs, typically

NVIDIA A100, which has got a large amount of VRAM and high performance in terms of FLPOS but at a very high price, beyond the budget of the most of the healthcare organizations. The average training time, per epoch, is about 54 minutes while the average inference time is about 45 seconds using greedy search decoding and 2 and a half minutes using beam search decoding.

We performed experiments combining different types of models and decoding types. These experiments show very promising results, especially combining the ViT input embedder and GPT-2, with an F1-score of 0.78 in case of Beam search. We would like to emphasise that the performance of this model can be increased by refining the dataset in different ways, such as collecting more data from different sources and testing different sampling strategies. Indeed, we employ a downsampling procedure on the entire dataset to rectify the imbalance in disease occurrences. Conversely, the utilization of data augmentation techniques, such as targeted cropping of specific regions in X-ray images, can bring advantages in terms of both raw performance metrics (e.g., BERTScore) and the model's generalization capacity. Furthermore, errors committed by the system in localizing some of the findings (e.g. a clinical condition located in the upper lobe of a lung is described by the system as "lower lobe"), could be reduced by increasing the dataset using the aforementioned cropping method.

As a prospective for further research, we intend to submit our computer-generated reports to human experts in order to validate the former through evaluation templates, such as SUS questionnaires[21], aimed at assessing their practical usability within clinical settings as supportive tools for healthcare professionals.

References

- [1] O. (2023), Gpt-4 technical report, arXiv:2303.08774 (2023).
- [2] M. Moor, Q. Huang, S. Wu, M. Yasunaga, C. Zakka, Y. Dalmia, E. P. Reis, P. Rajpurkar, J. Leskovec, Med-flamingo: A multimodal medical few-shot learner (2023). URL: <https://arxiv.org/abs/2307.15189>, arXiv:2307.15189.
- [3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in neural information processing systems, 2017, pp. 30–30.
- [4] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pretrained biomedical language model for biomedical text mining, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2020, pp. 4571–4580.
- [5] K. Huang, J. Altosaar, R. Ranganath, J. Leskovec, Clinicalbert: Modeling clinical notes and predicting hospital readmission, arXiv preprint arXiv:1904.05342 (2019).
- [6] Y. Gu, R. Tinn, H. Cheng, Y.-A. Cheng, R. Sekar, Y. Zhang, Y. Liu, W. Dai, Q. Qu, T. Walker, et al., Pubmedbert: A pretrained language model for biomedical text mining, arXiv preprint arXiv:2105.07774 (2021).
- [7] Y. Luo, Q. Song, Z. Yang, Y. Zhang, Biogpt: A general purpose language model fine-tuned on biomedical text, arXiv preprint arXiv:2201.05493 (2022).

- [8] A. Singhal, A. Banerjee, R. Sood, Med-palm: A large multimodal pre-trained language model for medical applications, arXiv preprint arXiv:2104.03495 (2021).
- [9] A. Awadalla, I. Gao, J. Gardner, J. Hessel, Y. Hanafy, W. Zhu, K. Marathe, Y. Bitton, S. Gadre, J. Jitsev, S. Kornblith, P. W. Koh, G. Ilharco, M. Wortsman, L. Schmidt, Openflamingo, 2023. URL: <https://doi.org/10.5281/zenodo.7733589>. doi:10.5281/zenodo.7733589.
- [10] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, K. Simonyan, Flamingo: a visual language model for few-shot learning, ArXiv abs/2204.14198 (2022).
- [11] A. E. Johnson, T. J. Pollard, S. J. Berkowitz, N. R. Greenbaum, M. P. Lungren, C. Y. Deng, R. G. Mark, S. Horng, Mimic-cxr: A large publicly available database of labeled chest radiographs, arXiv preprint arXiv:1901.07042 (2019).
- [12] M. Endo, R. Krishnan, V. Krishna, A. Y. Ng, P. Rajpurkar, Retrieval-based chest x-ray report generation using a pre-trained contrastive language-image model, in: Proceedings of Machine Learning for Health, volume 158 of *Proceedings of Machine Learning Research*, 2021, pp. 209–219.
- [13] P. R. Vignav Ramesh, Nathan Andrew Chi, Improving radiology report generation systems by removing hallucinated references to non-existent priors, in: arXiv:2210.06340, 2022.
- [14] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, Language models are unsupervised multitask learners (2019).
- [15] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, et al., Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning, arXiv preprint arXiv:1711.05225 (2017).
- [16] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, Z. Yan, M. Tomizuka, J. Gonzalez, K. Keutzer, P. Vajda, Visual transformers: Token-based image representation and processing for computer vision, 2020. arXiv:2006.03677.
- [17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE conference on computer vision and pattern recognition, Ieee, 2009, pp. 248–255.
- [18] T. Zhang, V. Kishore, F. Wu, K. Weinberger, Y. Artzi, Bertscore: Evaluating text generation with bert, in: Proc. 8th International Conference on Learning Representations (ICLR20), 2020. URL: <https://openreview.net/pdf?id=SkeHuCVFDr>.
- [19] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, Bertscore: Evaluating text generation with bert, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020.
- [20] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. arXiv:1810.04805.
- [21] J. Brooke, Sus: A quick and dirty usability scale, Usability Eval. Ind. 189 (1995).