Summarizing the content of Electronic Health Records and medical reports with a Large Language Model and Vision Language Model-Based processing of data

Anonymous ACL submission

Abstract

From the arrival of patients at a health facility to their discharge, a vast amount of highly valuable medical data are collected and gathered in Electronic Health Records (EHRs). However, current data management faces a certain number of limitations, linked with the amount and type of data used (tables, reports, images, etc.), that could hinder the efficiency of medical services. As a consequence, the analysis of records could be long and laborious for a medical personnel member, whereas the admission of patients in emergency situations calls for efficiency.

This paper presents a flexible Generative Artificial Intelligence-based framework for the processing of EHRs data. Through the use of Large Language Models and Vision Language Models, medical data are analyzed and aggregated in a single document summarizing the key information of a patient based on his/her medical history. This multimodal framework takes advantage of the strengths of language models to process structured data, medical reports, and medical images using text analysis, images processing, and Optical Character Recognition (OCR).

Experiments, conducted using hospital EHRs data from the Medical Information Mart for Intensive Care IV (compiling data from Beth Israel Deaconess Medical Center, Boston). and Language Models (including Mistal, Deepseek, LLaMA, Gemma, and LLaVA models) executed locally for medical data confidentiality, underscore promising results towards automated mutimodal processing of EHRs through summarization of reports in summaries 11 times shorter (for best LLMs) and the generation of image description with an extraction of texts with OCR.

1 Introduction

Medical data is at the center of the organization of health facilities. The exploitation of such data, providing valuable information about patients' current situations and reflecting their medical histories, is essential for decision-making and management of medical assets. In order to exploit efficient and easy-to-use solutions, Electronic Health Records (EHRs) are used by many health facilities to compile patient data. Such solutions are at the center of the activities of medical staff members who consult existing data and collect new information during the care of a patient.

However, medical data processing faces a certain number of challenges related to the amount and type of data used in health facilities. In fact, because data from previous admissions are kept in the patient history, the study of records of a patient with a lot of previous stays can be long for a medical personnel member whereas it is essential to correctly understand the patient background. The processing of medical data is also complicated by the type of data collected, including numerical data (such as vital signs) but also images (such as CT scan results) and textual data (medical reports, commentaries written by medical personnel, etc.).

This study focuses on processing the data present in EHRs to generate a summary of all the information available about a patient. The purpose of the developed framework is to aggregate all available data about a patient into a global summary which can be consulted by a member of the medical staff before a future admission of the same patient. Data processed include structured records, but also medical reports and images.

This article is structured as follows. Section 2 analyzes studies focusing on the processing of medical reports and the generation of medical summaries. Section 3 presents the approach developed by our team to generate medical summaries from EHRs, including the preparation of data and the processing of medical texts, images, and records. Section 4 evaluates and compares the different solutions . Section 5 draws a conclusion with prospects

and Section 6 discusses about the limitations of the work.

2 Literature review

Many studies have focused on developing NLP strategies to process EHRs data, extract interesting features, and provide insights.

First, approaches, such as (Vashishth et al., 2021), relying on Deep Learning models, were developed to process medical texts and provide data extraction with semantic analysis.

The latest advances in Generative Artificial Intelligence (Gen IA) and the advent of Large Language Models (LLMs) have shed light on generation techniques such as Retrieval-Augmented Generation (RAG), exploiting the capabilities of LLMs to produce data insights from external knowledge bases (Gao et al., 2023). Opportunities for data management, information retrieval were identified, but studies also raised challenges related to models and data management, such as (Yu et al., 2023) highlighting the necessity to develop inclusive solutions while dealing with data privacy/security and ethical concerns. Concrete applications of RAG were developed to process structured data and generate insights through Table-to-Text generation (Wu et al., 2022). RAG was also adapted to the processing of texts written in natural language (such as medical reports) for the summarization of data (Goswami et al., 2024) and the extraction of key information (Alkhalaf et al., 2024). In an effort to valorize the available data, generation solutions, based on language models (Handa et al., 2023) and transformer encoder–decoder architectures (Bazi et al., 2023), were introduced to adapt EHR data processing to image analysis.

Finally, the interactions between and users and data were developed through the creation of query-based generators (Zhao et al., 2023) leveraging the capabilities of Language Models to respond to users' questions from provided data and instructions. Medical applications of such technologies include chatbots that provide recommendations from patients' queries and information (Yang et al., 2024).

In summary, the studies in this literature review leverage RAG and Gen AI solutions to develop Data-to-Text solutions. Data are processed using various models in order to generate data descriptions, insights and summaries. However, they usually focus on the processing of one type of data whereas EHRs are made of heterogeneous data. A lot of Language Models-based studies also relies on the use of Open AI GPT models, raising questions about data confidentiality and computation capabilities required to run the solutions. In response to these challenges, this paper tries to develop a framework for multimodal processing of EHR data, summarized in a report that can be quickly reviewed by a member of the medical staff. The proposed solutions rely on open-source Language Models with few weights to process locally structured data, medical reports (texts), and medical images.

3 Methodology

3.1 Data exploited

This study, focusing on the processing of EHRs data, was carried out through the case of study of medical data collected in hospitals. The study was conducted using data from the Medical Information Mart for Intensive Care IV (MIMIC IV) [1] dataset (see (Johnson et al., 2023), (Johnson et al., 2020), and (Goldberger et al., 2000)), a public de-identified dataset that regroups data from the EHRs of the Beth Israel Deaconess Medical Center (Boston).

The experiments were conducted by filtering the data from four successive stays of two selected patients (based on the total number of stays registered in the database). In order to experiment with the generation of summaries, a certain number of parameters are kept, including patient-related data, hospital stay-related data, diagnoses and procedures for which the patient was billed during his/her hospital stays, and discharge reports produced after each hospital stay. All textual data are written in English. The data used for the experiment include both structured and unstructured data.

In addition, a medical image dataset was created to complete the records using publicly available images from an image bank[2]. Thus, a set composed of scanner result images with different sizes (to test the flexibility of the solutions) was created [3]. Additionally, vial pictures were taken and used to evaluate models on images containing texts and experiment OCR.

[1] MIMIC IV: https://mimic.mit.edu/

[2] Image bank : https://pixabay.com/

[3] Note: As this study aims to build a flexible framework and summarized data, the choice has been made to use various images (such as scan results) although they may not be related to the clinical condition of the patients studied in the experiments.

Data were filtered and aggregated to construct a data architecture, presented in Figure 1, which was used to simulate data management in hospitals and carry out experiments.

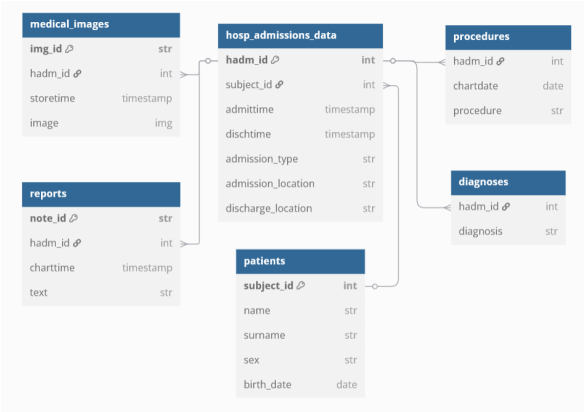

Figure 1: Architecture of the database used for experimentation.

Table 1 summarizes the number of records used for each type of data in the database.

Data	number of records
Patients	2
Hospital stays	8 *(4 for each patient)*
Diagnoses billed	64
Procedures billed	7
Medical reports	6
Medical images	7

Table 1: Number of records for each data exploited for experiments.

3.2 Processing of reports and texts

This section presents the approach developed to process texts written in natural language. This approach is used to process discharge reports and medical commentaries written by medical personnel to produce a brief textual summary of the main information of each document.

The solution experimented relies on the use of a Large Language Model (LLM) coupled with with the EHR reports in order to produce short summaries (50 to 80 words) of each document with key information. The prompt sent to the LLM is fueled by the content of each document in order to generate a summary with patient data.

In order to ensure the traceability of information, reports are processed one by one and summarized individually (with a prompt sent for each report) before all summaries are aggregated into a single document. This approach allows an easy identification of the document associated with each information and an easy sorting of data (using the date of each document) from the latest information to the oldest. The global architecture of this processing is summarized in Figure 2.

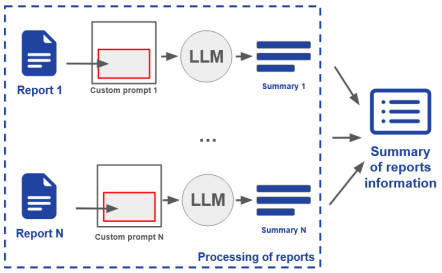

Figure 2: LLM-based processing of texts.

This approach is tested in section 4.2 which evaluates different LLMs used for the generation of summaries from reports.

3.3 Processing of images

Similarly to the previous section, this section details an approach used to summarize the content of images stored in the EHR.

Such processing is enabled by the use of Vision Language Model (VLM), a class of models able to generate texts from images and instructions. Hence, a VLM is fueled with each image of the EHR and is prompted to briefly describe the content of each image. This allows for a flexible processing of images (independent of the size of the image) and the automated generation of a list of images with their purpose. The OCR capabilities of VLM can also be used to enhance the content of the description (for example, to read the content of a label).

Similarly to the processing of texts, images are processed one by one and summaries are aggregated in a final result to provide a list of figures in the report detailing the content of medical images available.

This approach is tested in section 4.3 which evaluates different VLMs used for the generation of descriptions from images.

3.4 Processing of structured data

Structured data (list of diagnoses, procedures, etc.) are processed and summarized to complete the information provided by reports.

Similarly to the processing of text, Language Models are used to generate a textual description of

structured data using a LLM prompted with patient data. The use of LLM offers flexibility in final description produce and allows advanced processing of data such as grouping of information, conversion of units, or modification of the format of data (such as dates, presented in the dataset with a format "YYYY-MM-DD" but processed by the LLM to be presented with a format "Month date, Year"). The enhancement of the model with external data could also allow more complex processing, such as the processing of information encoded using specific standards (such as the Internal Classification of Diseases).

3.5 Aggregation of data

Once the EHR data are processed with the different methods presented in the previous sections, the data can be further processed and aggregated into a single document to summarize the key information for each patient.

A final document aggregates the summaries in a final document following the scheme of Figure 3.

Figure 3: Structure of the report summarizing patient data.

4 Computational results

4.1 Experimental protocol

The summarization and aggregation of EHRs data is evaluated using the data from the dataset (see Section 3) processed using the Pandas[4] Python library. Based on the records of each admission, summaries of data will be performed by different Language Models.

In an effort to build medical solutions respectful of the confidentiality of medical data and implementable in an architecture with limited computation capabilities, tests will be performed using LLMs [5] [6] [7] and VLMs [8] [9] [10] executed locally using Python and Ollama[11]. Tables 2 and 3 present the models used for evaluation.

LLM	Year	Weights *	License
Deepseek R1	2025	14B	*mit*
LLaMA 3.1	2024	8B	*llama 3.1*
Mistral	2023	7B	*apache-2.0*

Table 2: LLMs used for the summarization of reports and structured data.
* *Number of weights are expressed in billions.*

VLM	Year	Weights *	License
Gemma 3	2025	12B	*gemma*
LLaMA 3.2 vision	2024	11B	*llama3.2*
LLaVA	2023	13B	*apache-2.0*

Table 3: LLMs used for the summarization of reports and structured data.
* *Number of weights are expressed in billions.*

Thus, the tests will be conducted using a laptop equipped with 16 Go of Random-Access Memory (RAM), a Central Processing Unit (CPU) "Intel(R) Core(TM) i5-10300H CPU @ 2.50GHz", and a dedicated Graphics Processing Unit (GPU) "NVIDIA GeForce GTX 1650".

4.2 Evaluation of the performances of report summarization

4.2.1 Computation time needed to process a report

LLMs are first evaluated through a measurement of the computation time needed to process a medical report using the discharge reports of the dataset. The reports studied have a length between 923 and 1550 words. LLMs are prompted to summarize 5 of the 6 reports in around 50 to 80 words. The results are summarized in Figures 4 and 5.

[4] Library : https://pandas.pydata.org/

[5] Deepseek R1 model card: https://huggingface.co/deepseek-ai/DeepSeek-R1

[6] LLaMA 3.1 8B model card: https://huggingface.co/meta-LLaMA/LLaMA-3.1-8B

[7] Mistral 7B v0.3 model card: https://huggingface.co/mistralai/Mistral-7B-v0.3

[8] Gemma 3 model card: https://huggingface.co/google/gemma-3-12b-it

[9] LLaMA 3.2 Vision model card: https://huggingface.co/meta-LLaMA/LLaMA-3.2-11B-Vision

[10] LLaVA model: https://github.com/haotian-liu/LLaVA

[11] Ollama : https://ollama.com/ (MIT License)

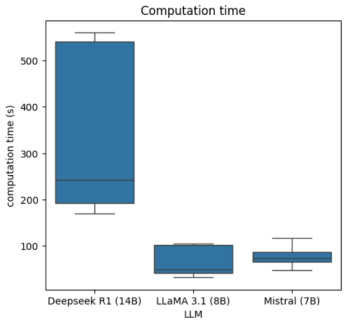

Figure 4: Computation times measured for the summarization of a discharge report with each LLM.

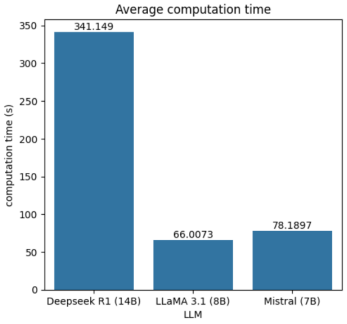

Figure 5: Average time needed by each LLM to summarize a report.

Experiments on discharge reports show that the evaluated models require 1mn to 5mn45s (Figure 5) to prepare a custom prompt from medical data and summarize a provided report. Figures underline that LLaMA 3.1 (8B) and Mistral (7B) tend to globally have a lower computation time (with average computation times between 1mn and 1mn15s per report) whereas Deepseek R1 needs more computation time before generating a final summary (as the generation is preceded by a "thinking" phase).

4.2.2 Evaluation of the summaries generated by the LLMs

The summaries of reports are evaluated using the metrics of the Recall-Oriented Understudy for Gisting Evaluation (ROUGE), a set of metrics to evaluate the quality of a summary comparing to a reference (Alkhalaf et al., 2024). The original report is used as reference and calculations are made using Paul Tardy's Python implementation [12].

The average results, using the evaluation of metrics for each report, are summarized in Tables 4, 5, and 6. The models are designated as follows: (1) refers to LLaMA 3.1 8B, (2) designates Mistral 7B, and (3) stands for Deepseek R1 14B.

ROUGE-1			
LLM	**recall**	**precision**	**F1-Score**
(1)	10.61%	66.18%	17.85%
(2)	16.08%	67.18%	25.28%
(3)	8.42%	52.89%	14.09%

Table 4: ROUGE-1 average metrics obtained by evaluating the models on 5 reports.

ROUGE-2			
LLM	**recall**	**precision**	**F1-Score**
(1)	4.01%	34.43%	7.0%
(2)	6.8%	34.47%	11.04%
(3)	2.02%	20.4%	3.62%

Table 5: ROUGE-2 average metrics obtained by evaluating the models on 5 reports.

ROUGE-L			
LLM	**recall**	**precision**	**F1-Score**
(1)	10.17%	63.61%	17.13%
(2)	15.35%	64.09%	24.12%
(3)	8.04%	50.56%	13.45%

Table 6: ROUGE-L average metrics obtained by evaluating the models on 5 reports.

In addition, the lengths of generated summaries are also evaluated using the spaces of the generated text. An average ratio between the lengths of the original reports and the lengths of the summaries is also calculated to measure the compression of information. The results are summarized in Figure 7.

The evaluation of the lengths of the summaries (Table 7) underscores the advantages of using LLMs to process reports and produce a shorter summary (with lengths divided by 6 to 12 on average). However, we can also notice that the instruction of summarizing each report in around 50 to 80 words is often not respected (especially by Mistral 7B).

In addition, the evaluation of the ROUGE metrics provides information on the ability of LLMs to summarize the original reports. The precision values, between 50.56% and 67.18% for ROUGE-1 and ROUGE-L, quantifying the ability of models to not generate false information, indicate that most of the words present in the summaries can be found in the original report. However, the value of recalls, indicating the ability of LLMs to preserve all the

[12] ROUGE Python library: https://github.com/pltrdy/rouge

	Report*	(1)	(2)	(3)
Report 1	1550	249	287	274
Report 2	1164	229	176	280
Report 3	1048	66	315	54
Report 4	925	49	90	67
Report 5	923	71	190	83
Average	1122	133	212	152
STD**	196	76	70	90
Average ratio	1.0	11.8	6.1	10.8

Table 7: Evaluation of the number of words of the summaries generated by LLMs.
(number of words of the original report).*
*(** standard deviation).*

information from the original report, also sheds light on the loss of information during the operation of summarization. Nevertheless, low recall values should be qualified by the fact that computations were performed using the original report as a reference instead of using a human-written summary (thus the difference of lengths between the two texts compared lowers the recall).

A visualization of examples of LLM-generated summaries (Figures 6 and Figures 7) sheds light on strengths and weaknesses of the approach.

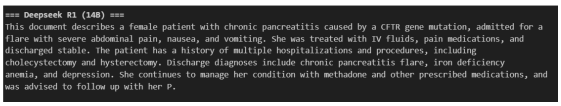

Figure 6: An example of summary computed by Deepseek R1 1(14B).

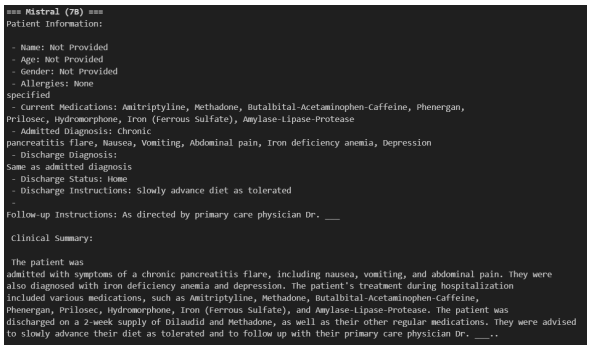

Figure 7: An example of the same summary computed by Mistral 7B.

The example presents two summaries that provide brief information from the discharge report of a patient. As previously seen, the summary produced by Mistral 7B is longer but provides further information, whereas the one provided by Deepseek R1 (14B) is more brief. We can also notice that the second summary keeps a structure with bullet points (similarly to the original report) whereas the first summary rephrases information.

However, the processing of long reports by low-weights models and a basic prompting strategy faces limitations related to model hallucinations that might lead to inaccurate data. For example, the report of 7 mentions that the patient's gender and allergies were not specified while the original result provided such information (although name and age were removed during de-identification). Another example is presented in Figure 8, which presents a case where the LLM did not generate a summary.

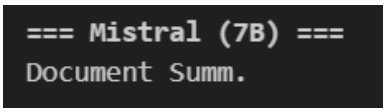

Figure 8: An example of an error in the generation of a summary by Mistral 7B.

4.3 Evaluation of the performances of images description

4.3.1 Computation time needed to process an image

Similarly to the previous evaluation, the computation time of VLMs is evaluated by measuring the time needed by each model to generate the description of a medical image using test images. VLMs are prompted to create a description with a length between 50 and 80 words. The results are summarized in Table 8.

	Gemma 3	LLaMA 3.2	LLaVA
average	119s	163s	29s
median	115s	161s	26s
min	113s	158s	22s
max	143s	180s	53s
STD*	10s	7s	10s

Table 8: Evaluation of the time needed by each VLM to process a medical image.
* *standard deviation*

From all the VLMs tested, LLaVA (13B) stands out for its capacity to process an image in around 30s, whereas the other models tested obtained image computation times between 2mn (Gemma 3 13B) and 2mn45 (LLaMA 3.2 vision 11B).

4.3.2 Study of the descriptions generated by the VLMs

The generated descriptions are analyses to evaluate the performance of each VLM in the processing of images. Figure 9 presents two examples of images processed with the output generated by each model.

Figure 9: An example of image processed with the description generated by each VLM.

In general, the models evaluated show promising results in the generation of short descriptions from images. The evaluation of models also underlines that Gemma 3 tends to produce brief descriptions, whereas LLaMA 3.2 Vision and LLaVA provided longer descriptions. This is especially the case for LLaVA which tends to produce longer descriptions by adding assumptions (which are sometimes incorrect).

OCR capabilities of VLM were also evaluated through the generation of descriptions for images containing texts. However, in an effort to build a flexible system capable of processing images with various styles, the prompt sent to each VLM was not modified to mention the presence of text in the image. Figures 10 and 10 present two examples of images used to test OCR capabilities of VLMs.

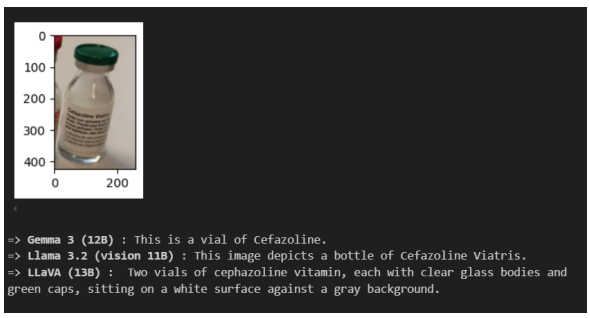

Figure 10: An example of image with texts processed.

The evaluations on images with texts underline that LLaVA has limited capabilities when it comes to reading label content without detailed instructions. For example, the model misspells the name of the vial in the results presented in 10 and did not summarize the content of the label presented in 11. LLaMA 3.2 Vision demonstrates the best performance in summarizing the content of vial labels with, for example, a description of the content of the label of Figure 11. We can notice in this example that the model used the name "Ceftazidime" whereas the first line of the label indicates "Ceftazidim" because the name "Ceftazidim(e)" also appears in the image.

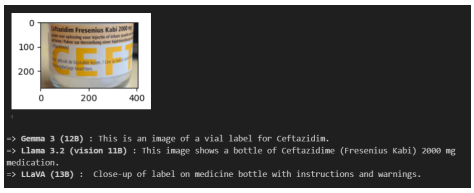

Figure 11: An example of image with texts processed.

Overall, the VLMs tested showed different strengths and weaknesses, including a VLM with low computation capabilities, summarizing the global content of an image but inclined to add assumptions (LLaVA), a VLM requiring more computation time but providing more detailed descriptions (LLaMA 3.2 Vision), and a VLM with intermediate computation times producing brief descriptions (Gemma 3). Depending on the usage, those solutions might be interesting and could be improved with a more precise prompt indicating information to describe/extract.

4.4 Evaluation of the performances of structured data processing

4.4.1 Computation time needed to process the structured data of an admission.

In this section, we measure the time needed by LLaMA 3.1 8B (1) and Mistral 7B (2) to process all procedures (billed) and diagnoses (billed) received during a hospital stay using data from multiple stays. The computation times are regrouped per table and represent times needed to process the data from the tables for one hospital stay. The average numbers of records per stay are $\simeq 1$ procedure per stay and $\simeq 8$ billed diagnoses per stay.

The average computation times are summarized in Table 9.

The calculation times measurement underline computation time of around 11s to 17s for the summarization of the procedures or the diagnoses of a hospital stay.

		procedures	diagnoses
(1)	average	11.4s	16.8s
	median	12.1s	16.4s
	min	6.0s	12.0s
	max	17.7s	23.3s
	STD*	3.9s	3.5s
(2)	average	12.7s	16.8s
	median	9.0s	16.9s
	min	6.5s	12.0s
	max	31.9s	22.0s
	STD*	8.8s	3.8s

Table 9: Evaluation of the time needed by each LLM to process the data of a hospital stay.
* *standard deviation*

4.4.2 Study of the summaries of structured data generated by the LLMs

A first analysis of the summaries of billed diagnoses and procedures is performed. Some examples of structured data with the summaries generated by LLMs are provided in Figures 12 and 13.

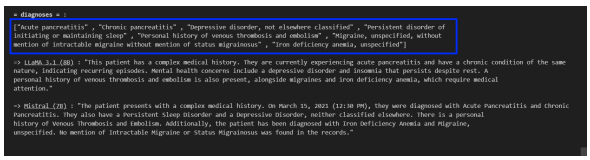

Figure 12: An example of generated summary using the list of diagnoses identified during a stay (data retrieved from the database are framed in blue).

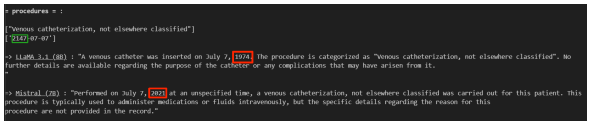

Figure 13: An example of generated summary for the list of procedures during a stay (with dates).

Similarly to the observations made during the summarization of reports (see Section 3.2), the tests show that Mistral 7B and LLaMA 3.1 8B globally succeed in generating a description from the diagnoses and procedures. A conversion of dates was also performed by the LLMs to present dates in a "Month Date, Year" format, while dates are stored in the database following the ISO 8601 format ("YYYY-MM-DD").

However, LLM hallucinations were also identified in certain situations, such as when no procedures are provided (see Figure 14), leading to a summary of data with false information. Similarly, models modified the years in each procedure date (see the years framed in red). This might be linked to the fact that the year was shifted to 2174 for de-identification.

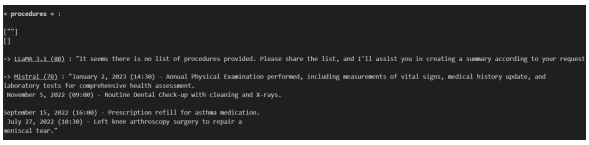

Figure 14: An example of hallucination from Mistral 7B when no procedures are provided.

In general, the use of LLM should be adapted to perform an efficient text summarization providing brief and precise information. As using a prompt with general instructions shows limited performances, adapting a prompt for each type of data could be an interesting approach for a more accurate and customized description tables.

5 Conclusion

This works enables the creation of a framework for the processing and summarization of EHR data. It tackles the complexity of images and unstructured data through a transversal approach relying on Language Models. Evaluations on hospital EHRs data underscore the potential of LLMs and VLMs in the processing of complex data, as models are able to produce synthetic summaries using information from images, reports, and tables to provide an overview on the medical history of a patient. However, limited computation capabilities of low-weights models (executed locally) induce challenges in computation times and accuracy of summaries.

Opportunities for the improvement of the proposed architecture include the implementation of more advanced prompt strategies to control and correct the data generated by language models. The specialization of open-source language models in collaboration with medical experts could also increase the quality of summaries generated while taking advantage of the confidentiality of data and lower energy consumption provided by local low-weights models.

6 Limitations

This study was conducted using a certain number of assumptions implied by the data and models manipulated. In fact, although the methodology

developed is thought to be transversal and adaptive, tests were conducted on a limited number of data from a few databases. This is especially the case for medical reports that were taken from the same database and thus have a similar structure and the same language (English). More detailed tests with more data and the evaluation of an expert could be beneficial to enhance the proposed architecture.

The construction of an automated tool for EHR data summarization could improve the work of medical personnel by making the retrieval and analysis of information easier. However, such a tool should be protected with rigorous control of data accessibility to avoid leakage of patient data and respect of medical secrecy. Generative models must also be safeguarded to avoid the generation of unintended content and rigorously evaluated to ensure the veracity of the information provided. Ignoring such controls could hinder the admission of patients (because of incorrect information) and allow misuse of the solution to retrieve confidential medical data.

References

Arthur Mensch Chris Bamford Devendra Singh Chaplot Diego de las Casas Florian Bressand Gianna Lengyel Guillaume Lample Lucile Saulnier Lélio Renard Lavaud Marie-Anne Lachaux Pierre Stock Teven Le Scao Thibaut Lavril Thomas Wang Timothée Lacroix William El Sayed Albert Q. Jiang, Alexandre Sablayrolles. 2023. Mistral 7b. *arXiv preprint arXiv:2407.21783*.

Mohammad Alkhalaf, Ping Yu, Mengyang Yin, and Chao Deng. 2024. Applying generative ai with retrieval augmented generation to summarize and extract key clinical information from electronic health records. *Journal of biomedical informatics*, 156:104662.

Yakoub Bazi, Mohamad Mahmoud Al Rahhal, Laila Bashmal, and Mansour Zuair. 2023. Vision–language model for visual question answering in medical imagery. *Bioengineering*, 10(3):380.

Jianfeng Chi, Ujjwal Karn, Hongyuan Zhan, Eric Smith, Javier Rando, Yiming Zhang, Kate Plawiak, Zacharie Delpierre Coudert, Kartikeya Upasani, and Mahesh Pasupuleti. 2024. Llama guard 3 vision: Safeguarding human-ai image understanding conversations. *arXiv preprint arXiv:2411.10414*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2:1.

Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. 2000. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220.

Joyeeta Goswami, Kaushal Kumar Prajapati, Ashim Saha, and Apu Kumar Saha. 2024. Parameter-efficient fine-tuning large language model approach for hospital discharge paper summarization. *Applied Soft Computing*, 157:111531.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Palak Handa, Deepti Chhabra, Nidhi Goel, and Sri Krishnan. 2023. Exploring the role of chatgpt in medical image analysis. *Biomedical Signal Processing and Control*, 86:105292.

Alistair Johnson, Lucas Bulgarelli, Tom Pollard, Steven Horng, Leo Anthony Celi, and Roger Mark. 2020. Mimic-iv. *PhysioNet. Available online at: https://physionet. org/content/mimiciv/1.0/(accessed August 23, 2021)*, pages 49–55.

Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, and 1 others. 2023. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Shikhar Vashishth, Denis Newman-Griffis, Rishabh Joshi, Ritam Dutt, and Carolyn P Rosé. 2021. Improving broad-coverage medical entity linking with semantic type prediction and large-scale datasets. *Journal of biomedical informatics*, 121:103880.

Heng-Yi Wu, Jingqing Zhang, Julia Ive, Tong Li, Vibhor Gupta, Bingyuan Chen, and Yike Guo. 2022. Medical scientific table-to-text generation with human-in-the-loop under the data sparsity constraint. *arXiv preprint arXiv:2205.12368*.

Zhongqi Yang, Elahe Khatibi, Nitish Nagesh, Mahyar Abbasian, Iman Azimi, Ramesh Jain, and Amir M Rahmani. 2024. Chatdiet: Empowering personalized nutrition-oriented food recommender chatbots through an llm-augmented framework. *Smart Health*, 32:100465.

Ping Yu, Hua Xu, Xia Hu, and Chao Deng. 2023. Leveraging generative ai and large language models: a comprehensive roadmap for healthcare integration. In *Healthcare*, volume 11, page 2776. MDPI.

Yilun Zhao, Haowei Zhang, Shengyun Si, Linyong Nan, Xiangru Tang, and Arman Cohan. 2023. Investigating table-to-text generation capabilities of llms in real-world information seeking scenarios. *arXiv preprint arXiv:2305.14987*.