

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Generating Synthetic Clinical Data in European Portuguese Using an Open-Source Large Language Model

Daniel Filipe Souto Félix



Mestrado em Engenharia Informática e Computação

Supervisor: Prof. Carla Alexandra Teixeira Lopes

July 27, 2025

Generating Synthetic Clinical Data in European Portuguese Using an Open-Source Large Language Model

Daniel Filipe Souto Félix

Mestrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

President: Prof. Mariana Malta

Referee: Prof. Ricardo Campos

July 27, 2025

Resumo

Tirar partido de avanços recentes nas tecnologias clínicas de Inteligência Artificial (IA) depende muito da disponibilidade de *datasets* de dados clínicos de alta qualidade. No entanto, dados clínicos reais são difíceis de obter devido a questões de privacidade e restrições legais. Como tal, línguas como o português europeu ficaram para trás em termos de investigação. Dados sintéticos têm sido investigados recentemente como um meio de substituir dados reais no desenvolvimento de modelos de IA e na investigação em geral. Esta dissertação investiga a viabilidade de gerar um *dataset* de notas clínicas sintéticas em português europeu usando um modelo de linguagem de grande escala de código aberto, com o objetivo de mitigar as limitações desta língua e contribuir para uma investigação clínica de IA mais inclusiva.

Foi desenvolvida uma abordagem completa para a geração, anotação e avaliação de um *dataset* de notas clínicas sintéticas em português europeu, ao qual demos o nome de LusoClin. A geração incluiu a seleção e o pré-processamento de dados clínicos reais para auxiliar na geração de dados, *fine-tuning* do modelo Llama 3.2 (com 3 mil milhões de parâmetros e *instruction-tuning*) utilizando *Quantized Low-Rank Adaptation*, e engenharia de *prompts* para gerar as notas clínicas. As notas sintéticas resultantes foram automaticamente anotadas usando um modelo BERT e avaliadas em quatro características: utilidade, fidelidade, diversidade e privacidade. A avaliação foi realizada utilizando métodos humanos e automáticos.

Os resultados indicam que, embora o LusoClin tenha um valor razoável, a qualidade das notas clínicas sintéticas tem muito espaço para melhorias. Os resultados dos métodos de avaliação automáticos foram promissores, mas os avaliadores humanos foram frequentemente capazes de distinguir as notas sintéticas das reais, e inconsistências estruturais e semânticas, bem como erros médicos, foram recorrentes. Não obstante, o LusoClin tem potencial para ser útil no desenvolvimento de modelos clínicos de IA em português europeu e em qualquer outra investigação que requeira dados textuais clínicos não estruturados. Além disso, a abordagem que desenvolvemos e descrevemos para esta tarefa e a nossa investigação exaustiva sobre a capacidade do modelo utilizado para gerar conteúdo clínico em português europeu podem ser valiosas para futuras investigações de natureza semelhante.

O LusoClin e outros resultados relevantes desta dissertação estão disponíveis publicamente num repositório do INESC TEC [37]. Trabalhos futuros devem ter como objetivos experimentar diferentes modelos de linguagem de grande escala de código aberto, fortalecer metodologias de avaliação e explorar diferentes técnicas de *fine-tuning* e engenharia de *prompts* para melhorar a qualidade da produção sintética clínica.

Palavras-chave: Geração de Dados Clínicos Sintéticos, Modelo de Linguagem de Grande Escala, *Named Entity Recognition*, Aprendizagem Profunda, Notas Clínicas, BERT

Abstract

Taking advantage of recent advancements in clinical Artificial Intelligence (AI) technologies depends heavily on the availability of large, high-quality clinical datasets. However, real clinical data is difficult to obtain due to privacy concerns and legal restrictions. Due to this, languages such as European Portuguese have fallen behind in terms of research. Synthetic data has been investigated lately as a means of replacing real data in the development of AI models and research in general. This dissertation investigates the feasibility of generating a dataset of synthetic clinical notes in European Portuguese using an open-source large language model, with the goal of mitigating the limitations of this language and contributing to more inclusive clinical AI research.

A complete approach was developed for the generation, annotation, and evaluation of a dataset of synthetic clinical notes in European Portuguese, which we named LusoClin. Generation included the selection and pre-processing of real clinical data to assist in the data generation, fine-tuning the Llama 3.2 model (with 3 billion parameters and instruction-tuning) using Quantized Low-Rank Adaptation, and engineering prompts to generate the clinical notes. The resulting synthetic notes were automatically annotated using a BERT-based model and evaluated on four characteristics: utility, fidelity, diversity, and privacy. Evaluation was conducted using both human and automatic methods.

The results indicate that, while LusoClin holds reasonable value, the quality of the synthetic clinical notes has plenty of room for improvement. Results of automatic evaluation methods were promising, but human evaluators were frequently able to distinguish synthetic from real notes, and structural and semantic inconsistencies as well as medical errors were recurrent. Nevertheless, LusoClin still holds potential to be useful in helping the development of European Portuguese clinical AI models and any other research that requires unstructured clinical textual data. Furthermore, the approach we developed and described for this task and our extensive investigation into the used model's capacity to generate clinical European Portuguese content may be valuable for future investigations of a similar nature.

LusoClin and other relevant results of this dissertation are publicly available in an INESC TEC repository [37]. Future work should aim to experiment with different open-source large language models, strengthen evaluation methodologies, and explore different fine-tuning and prompt engineering techniques to improve the quality of the synthetic clinical output.

Keywords: Clinical Synthetic Data Generation, Large Language Model, Named Entity Recognition, Deep Learning, Clinical Notes, BERT

United Nations Sustainable Development Goals

The United Nations Sustainable Development Goals (SDGs) are a set of 17 global goals adopted by all UN Member States in 2015 as part of the 2030 Agenda for Sustainable Development. They are shown in [Figure 1](#). SDGs aim to address a wide range of global challenges, such as poverty, inequality, and climate change, and to ensure a better and more sustainable future for all by 2030 [4].



Figure 1: The 17 United Nations SDGs. Source: United Nations [4].

This dissertation project creates measurable impact related to United Nations SDGs, as detailed by [Table 1](#), which summarizes the goals to which this work contributed.

SDG	Target	Contribution(s)
3	3.8	Publicly available datasets in local languages can improve the quality and cultural applicability of Artificial Intelligence tools in healthcare, such as clinical decision support and clinical search engines, thus promoting access to quality and essential healthcare services.
	3.b	By creating a publicly available, synthetic clinical dataset, we're reducing the data access barriers in healthcare research and development, especially for underrepresented languages (European Portuguese, in this case). This helps development and validation of medical models, algorithms, or treatments in the target language, facilitating more inclusive healthcare innovation.
	3.c	Our dataset could help develop Artificial Intelligence models used to train healthcare professionals, especially in medical informatics, diagnosis assisted by Artificial Intelligence, and patient interaction tools — particularly in Portuguese-speaking contexts, which are lacking in datasets of this nature.
9	9.5	By developing a publicly available, synthetic clinical dataset in European Portuguese, our work supports scientific research in healthcare Artificial Intelligence. It enables experimentation, model training, and reproducibility in low-resource language contexts, fostering innovation and reducing geographic and linguistic barriers to development of clinical Artificial Intelligence tools.
	9.c	This project contributes to digital infrastructure by increasing the availability of language-specific clinical data for Artificial Intelligence development. This enhances access to medical information systems and language technologies for Portuguese-speaking people, supporting equitable access to health innovation.

Table 1: United Nations SDG contributions by this dissertation.

Agradecimentos

Em primeiro lugar, gostaria de deixar o meu agradecimento à minha orientadora, a professora Carla Teixeira Lopes, por toda a sua incansável disponibilidade, pelo seu inabalável apoio e pela sua paciência, que me guiaram durante o projeto e me seguraram nos momentos de maior dificuldade. A sua experiência e orientação foram os verdadeiros pilares desta dissertação.

À minha irmã, estou profundamente agradecido pelo amor, amizade, companhia e apoio emocional que, durante esta jornada e toda a minha vida, foram e são a minha luz e a força que me move.

Aos meus pais, quero expressar a minha gratidão pelo apoio incondicional fornecido durante esta etapa da minha vida e pela compreensão perante as dificuldades que fui encontrando e que estenderam este processo por mais tempo do que era esperado.

À Inês e à Luísa, quero agradecer a inestimável ajuda e conhecimento técnico, que tornaram possível uma das fases cruciais deste projeto.

A todos os alunos e professores de Medicina e profissionais médicos que colaboraram numa das fases mais importantes deste projeto, quero agradecer a sua disponibilidade e conhecimento, que acrescentaram imenso valor a esta dissertação e aos seus resultados.

Por último, quero agradecer ao Luís e ao Pedro pelo apoio e amizade durante a minha jornada académica. Sem vocês, certamente não teria chegado aqui. Os vossos incontáveis sacrifícios altruístas, a vossa disponibilidade e o vosso companheirismo tornaram tudo isto possível.

Daniel Félix

*“Falhei em tudo.
Como não fiz propósito nenhum, talvez tudo fosse nada.”*

Álvaro de Campos, Tabacaria

Contents

1	Introduction	1
1.1	Context	1
1.2	Motivation and Goals	2
1.3	Contributions	2
1.4	Dissertation structure	3
2	Background	4
2.1	Natural language processing	4
2.2	Information extraction	5
2.2.1	Tasks	5
2.2.2	Applications	8
2.2.3	Challenges	8
2.3	Neural networks	9
2.3.1	Convolutional Neural Networks	9
2.3.2	Recurrent Neural Networks	10
2.3.3	Transformers	11
2.3.4	Embeddings	12
2.4	Language Models	12
2.4.1	Large Language Models	12
2.4.2	Generative large language models	14
2.4.3	BERT	14
2.4.4	LLM-based information extraction	15
3	Synthetic clinical data generation	17
3.1	Synthetic data generation	17
3.1.1	Overview of methods	17
3.1.2	Applications	19
3.2	LLM-based generation of synthetic unstructured clinical textual data	20
3.2.1	Preparation	21
3.2.2	Prompt engineering	22
3.2.3	Evaluation	24
3.2.4	Datasets of synthetic unstructured clinical textual data	30
3.2.5	Challenges	30
3.3	Conclusions	30
4	Methodology	32
4.1	Approach	32
4.2	Preparation	34

4.2.1	LLM selection	34
4.2.2	Real dataset selection	35
4.2.3	Data pre-processing	40
4.2.4	Fine-tuning using QLoRA	40
4.3	Prompt engineering and dataset generation	42
4.3.1	Prompt engineering	42
4.3.2	Small-scale evaluation of prompt results	45
4.3.3	Dataset generation	49
4.3.4	Dataset post-processing	49
4.4	Annotation	50
4.4.1	LLM selection	50
4.4.2	NER Annotation	50
5	LusoClin evaluation	52
5.1	Evaluation	52
5.1.1	Human evaluation	52
5.1.2	Automatic evaluation	55
5.2	Results	56
5.2.1	Human evaluation	56
5.2.2	Automatic evaluation	61
5.3	Conclusions	66
6	Conclusions	67
6.1	Findings	67
6.2	Project limitations	68
6.3	Future work	69
	References	71
A	Engineered prompts	86

List of Figures

2.1	General domain NER example. Source: Jehangir et al. [64].	6
2.2	Medical NER example. Source: Mehmood et al. [94].	6
2.3	Clinical RE example showing drug-adverse effect relations. Source: Raza and Schwartz [115].	7
2.4	Clinical EL example. Source: Shi et al. [125].	7
2.5	Clinical EE example. Source: Li et al. [82].	8
2.6	CNN-based model for NLP. Source: Tam et al. [135].	10
2.7	Transformer architecture. Source: Vaswani et al. [147].	11
2.8	LLM capabilities. Source: Minaee et al. [100].	13
2.9	An example of a BERT embedding. Source: Devlin et al. [27].	14
2.10	Overall pre-training and fine-tuning procedures for BERT. Source: Devlin et al. [27].	15
3.1	ClinGen prompt engineering approach overview. Source: Xu et al. [158].	23
3.2	Diagram of ML model performance as an evaluation method. Source: Figueira and Vaz [32].	27
3.3	t-SNE plot example. Source: Xu et al. [158].	28
4.1	Architecture of the approach used to generate, annotate, and evaluate LusoClin. .	33
4.2	Llama 3 architecture. Source: Dubey et al. [140].	34
4.3	Different types of clinical notes have different word distributions.	37
4.4	Different types of clinical notes have different sentence distributions.	37
4.5	Word count distribution for CLINpt’s clinical cases.	39
4.6	Sentence count distribution for CLINpt’s clinical cases.	39
4.7	LoRA method. Source: Hu et al. [60].	41
4.8	Diagram of small-scale prompt evaluation.	46
5.1	Web application’s evaluation page.	54
5.2	Web application’s user profile page.	55
5.3	Word count distributions for each note type in LusoClin and SemClinBr.	61
5.4	Sentence count distributions for each note type in LusoClin and SemClinBr. . . .	62
5.5	Synthetic clinical notes have more entities than real notes.	63
5.6	Entity count distributions for each entity type in LusoClin and SemClinBr - Part 1.	63
5.7	Entity count distributions for each entity type in LusoClin and SemClinBr - Part 2.	64
5.8	t-SNE plots of LusoClin and SemClinBr.	64

List of Tables

2.1	Comparison between LLMs and other types of machine learning models. Source: Chang et al. [16].	12
3.1	Studies on generation of synthetic clinical data of various types.	19
3.2	Prompt engineering approaches used by various studies.	22
3.3	Evaluation methods for synthetic clinical data.	25
3.4	Datasets of synthetic unstructured clinical textual data.	31
4.1	LLMs and reasons for exclusion.	35
4.2	Number and percentage of notes for each medical specialty in the SemClinBr dataset. Medical specialties with fewer than ten occurrences were grouped into the “Others” category. Source: Oliveira et al. [104].	36
4.3	Annotated excerpt from CLINpt. Source: Lopes et al. [86].	39
4.4	Data pre-processing operations.	40
4.5	Conclusions about prompt challenges the model faced.	43
4.6	Prompt template after <i>prompt engineering</i> and an English translation.	44
4.7	Type guessing results.	47
4.8	Turing test results.	47
4.9	Evaluation parameter results.	48
4.10	MediAlbertina’s entity types.	50
5.1	Summary of evaluation methods used and dimensions they evaluate.	53
5.2	Evaluation distribution of the 71 clinical notes that were evaluated twice.	57
5.3	Human evaluation - Turing test results.	58
5.4	Ratios of concordant and discordant justification for why notes were evaluated as synthetic.	58
5.5	Summary of justification why notes were evaluated as synthetic. 24 LusoClin and 6 SemClinBr evaluations provided no justification.	60
5.6	Summary of comments made about selected text excerpts on notes evaluated as synthetic. 2 selected excerpts had no associated comment.	60
5.7	Perplexity results. Both models were better at predicting the next token in the synthetic data than in the real data.	61
5.8	Frequency of each entity type on LusoClin and SemClinBr.	62
5.9	Average cosine similarity between pairs of samples from the same dataset.	65
5.10	Average cosine similarity between pairs of one LusoClin and one SemClinBr note.	65
5.11	Average Hausdorff distance between pairs of one LusoClin and one SemClinBr note.	65
6.1	Summary of evaluation results, divided by dimension.	69

List of Acronyms

AI	Artificial Intelligence
SDG	Sustainable Development Goals
ML	Machine Learning
DL	Deep Learning
NLP	Natural Language Processing
IE	Information Extraction
LLM	Large Language Model
PoS	Part-of-speech
NER	Named Entity Recognition
RE	Relation Extraction
EL	Entity Linking
EE	Event Extraction
NN	Neural Network
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
BRNN	Bidirectional Recurrent Neural Networks
LSTM	Long Short-Term Memory
BiLSTM	Bidirectional-LSTM
GRU	Gated Recurrent Unit
BERT	Bidirectional Encoder Representations from Transformers
GloVe	Global Vectors for Word Representation
LM	Language Model
RLHF	Reinforcement Learning with Human Feedback
NSP	Next Sentence Prediction
GAN	Generative Adversarial Network
VAE	Variational Autoencoder
EHR	Electronic Health Record
TSTR	Train on Synthetic, Test on Real
TRTS	Train on Real, Test on Synthetic
TSTS	Train on Synthetic, Test on Synthetic
PPL	Perplexity
t-SNE	T-distributed Stochastic Neighbor Embedding
CMD	Central Moment Discrepancy
IOB	Inside, outside, beginning
QLoRA	Quantized Low-Rank Adaptation
LoRA	Low-Rank Adaptation
NF4	4-bit NormalFloat
DQ	Double Quantization

Chapter 1

Introduction

In this chapter, we contextualize this project, explain its motivation and goals, describe its contributions, and summarize the structure of the document.

1.1 Context

The integration of recent technological advancements, namely artificial intelligence (AI), into healthcare offers substantial benefits across several domains. Advancements in AI technologies, like machine learning (ML), deep learning (DL), natural language processing (NLP), and information extraction (IE) have lead to improvements in various medical tasks, such as early detection, patient treatment, clinical search engines, recruitment of patients for clinical trials, and treatment prescription with decision support, among others [63, 125].

Information extraction is the automatic extraction of relevant information from unstructured (natural language) text. Unstructured text is the most common form of clinical data [81]. This type of clinical data contains plenty of useful information, such as clinical observations, treatment narratives, and response to medication, that is costly to extract manually, as it requires medical professionals to invest time and effort in the task [125]. The advent of AI information extraction models, especially deep-learning models, has shown that automatically extracting this information is a viable path [44]. However, many of these models, called supervised, require large amounts of data for training, so they can learn how to properly perform their task. Furthermore, all models, supervised and unsupervised, require data for evaluation before being available for use in real scenarios. The challenge is that clinical data needs to be professionally curated before becoming available for these purposes, since it contains plenty of sensitive information that cannot be publicly shared and may need expert annotation to be useful [85]. The English language has seen a significant increase in the available medical *corpora* that meets the needs of the field [125], but the same cannot be said for other languages (such as Portuguese), which have more limited resources and, thus, have fallen behind in terms of research [120].

Synthetic data, which is computer-generated data used to augment or replace real data, has the potential to address and diminish this issue. The generation of synthetic data has recently garnered attention due to the advent of large language models (LLMs). Large language models are advanced computational models trained on massive amounts of data and capable of understanding and generating natural language [16]. They have been explored in synthetic data generation because they can considerably reduce the cost of the task, while maintaining quality [85], even in the case of synthetic medical information [77, 162, 165]. Therefore, synthetic data generation constitutes an avenue worth exploring, in an attempt to allow Portuguese and other low-resource languages to take advantage of the recent advances in clinical information extraction.

1.2 Motivation and Goals

With regard to the use of the latest techniques in the development of models for clinical NLP tasks, Brazilian Portuguese has received some attention in recent years [78, 111, 121]. However, the same cannot be said for European Portuguese [87], mainly due to a lack of high-quality clinical data publicly available for research [86]. Therefore, European Portuguese would greatly benefit from an increase in available clinical data. Furthermore, given that real clinical data is generated in the language of its country of origin, it is also useful to determine whether recent NLP techniques are effective in languages other than English.

The goal of this project is twofold: exploring the ability of open-source large language models to generate European Portuguese synthetic clinical data, which is of enough quality to be used in training and evaluating models for downstream clinical tasks and any other research that could benefit from it; producing and making publicly available a large-scale synthetic dataset of clinical notes in European Portuguese.

1.3 Contributions

Our main contribution is a publicly available, large-scale synthetic dataset of clinical notes in European Portuguese, named LusoClin. Although LusoClin has room for several major improvements, its current state has the potential to be valuable in aiding the development of European Portuguese AI models for downstream clinical tasks, as well as other research. This potential is greatly increased by the fact that European Portuguese is a very underresourced language in terms of publicly available clinical data.

Another contribution is an extensive analysis of Llama 3.2 with 3 billion parameters and instruction tuning’s ability to generate European Portuguese unstructured clinical textual data.

LusoClin, the prompts and responses from the prompt engineering process, and the results from the human evaluation are all available in a dedicated repository [37].

In addition, we are in the process of writing an article with the same title as this dissertation, ‘Generating Synthetic Clinical Data in European Portuguese Using an Open-Source Large Language Model’, which will contain the information, approach, results, and analyses found in this document.

1.4 Dissertation structure

Besides the Introduction, this dissertation contains five further chapters: in [Chapter 2](#), we explain background concepts crucial to this dissertation’s understanding; in [Chapter 3](#), we introduce synthetic data generation and review state-of-the-art methods of LLM-based generation of synthetic unstructured clinical textual data; in [Chapter 4](#), we present our research hypothesis and detail our approach to generate LusoClin, a synthetic dataset of clinical notes in European Portuguese; in [Chapter 5](#), we describe our evaluation of LusoClin and analyse its results; in [Chapter 6](#), we conclude this dissertation by analysing our findings and project limitations, and by proposing some guidelines for future work.

Chapter 2

Background

In this chapter, we begin by defining natural language processing and describing some low- and high-level tasks. Afterward, we define information extraction, presenting some of its tasks, applications, and challenges in the clinical domain. Next, we present neural networks as the foundation for recent approaches to NLP tasks, such as information extraction. We describe the main types of neural network architectures used in recent models. Lastly, language models are defined, with a special focus on the concept of LLM, finishing with a portrait of the use of LLMs in information extraction.

2.1 Natural language processing

Natural language processing (NLP) is an AI field that aims to understand text written in human languages [74]. In other words, NLP attempts to obtain meaning from highly unstructured human-written text [130].

NLP encapsulates various tasks, which can be divided into two main categories: low-level and high-level [102]. Low-level tasks are employed at the syntactic level, with the goal of preparing the human-written text for high-level tasks [128].

Low-level tasks target different goals and can be combined for better results [130]. We introduce some here. Tokenization consists of parsing the text into tokens, which can be words, sentences, or other text spans. Tokens can be processed individually or as a sequence [130]. The most basic form of tokenization is parsing the text by whitespaces and punctuation. However, as is the case in clinical text, specific domains tend to have morphologically complex entities in their vocabulary. For example, characters that are typically used as dividers, such as hyphens, can be part of a token, such as with the entity ‘N-acetylcysteine’ [102]. Stemming consists of normalizing a token to its base form, called *stem*, frequently involving the elimination of derivational affixes [92]. The stem can correspond to a word or a fragment of a word [130]. For example, the stem of ‘sweeter’ is ‘sweet’ and the stem of ‘having’ is ‘hav’. Lemmatization consists of normalizing a token in a way that considers vocabulary, morphology, and context, in order to remove inflectional endings and return a sensible base form, called *lemma* [92, 130]. A lemma always corresponds to a real

word. For example, the lemma of ‘having’ is ‘have’. Part-of-speech (PoS) tagging consists of assigning a category to each token, representing its role in the sentence [130]. Examples of categories are adverb, noun, and number. Morphological decomposition of compound words consists of parsing compound words into the lemmas that form them. It is especially useful in clinical text since clinical entities are heavy on compound words [102]. For example, ‘cardiopulmonary’ can be parsed into the lemmas ‘cardio’, ‘pulmo’, and a suffix. Problem-specific segmentation consists of segmenting text into relevant groups, turning unstructured text into semi-structured. For example, an admission note template may contain the segments ‘Chief Complaint’ and ‘Past Medical History’, among others [102].

Conversely, high-level tasks are used at the semantic level, with the goal of understanding the text and extracting meaning from it [128]. These take advantage of the pre-processing performed by low-level tasks and can support each other in high-level applications. These are a few examples. Information extraction consists of utilizing computational methods to identify relevant information in the text and represent it in a computer-friendly form, to be used for computer-based storage, processing, retrieval, and analysis [155]. Question answering consists of automatically answering questions asked in natural language [130]. Topic modelling consists of classifying text according to their topic(s), in order to more easily assess their relevance to a task [131]. Machine translation consists of automatically translating natural language text from one language to another [97]. Summarization consists of generating concise summaries of longer documents of natural text. This task acts as a bridge between the abundance of textual information and the necessity for succinct, informative representations [133]. Sentiment analysis consists of evaluating and identifying the emotional tone or mood behind textual data [67]. Speech recognition consists of automatically converting natural speech into textual data [21].

2.2 Information extraction

Information extraction (IE) is the automatic extraction of relevant information from unstructured text, in the form of entities, relations, objects, events, and others [8, 118]. This information is converted into machine-readable, structured representations so it can be stored, processed, retrieved, and analysed [155]. IE, as a task, encompasses all NLP tasks that extract any form of information from unstructured text, such as named entity recognition, relation extraction, entity linking, and event extraction [8, 39, 163].

2.2.1 Tasks

Any NLP task that aims to extract structured information from unstructured, natural language text is an IE task [163]. In this section, we highlight some of the most relevant IE tasks in the clinical domain.

Named entity recognition (NER) is an IE task consisting of identifying text spans, known as named entities, within unstructured text. Named entities are text spans that refer to real-world objects and concepts, such as people, locations, and dates [130]. In clinical NER, the goal is

to identify medical named entities, such as diseases, symptoms, and drugs [136]. NER can also be taken a step further by grouping the entities with labels such as person or geographical location [94]. In the clinical domain, it is typical to group them by generic medical types, like gene, drug, or disease [94, 131] or by a target knowledge base's semantic types [104]. Figure 2.1 depicts a NER example with general domain text and labels, while Figure 2.2 depicts one with medical text and labels.

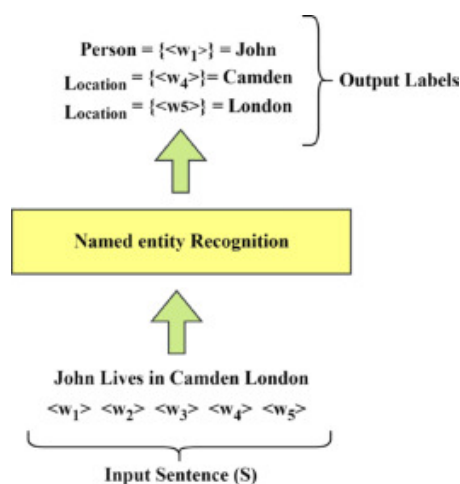


Figure 2.1: General domain NER example. Source: Jehangir et al. [64].

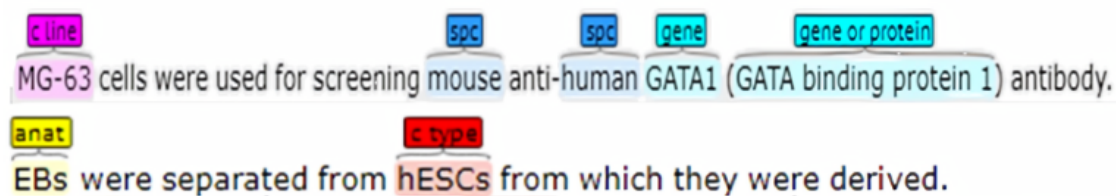


Figure 2.2: Medical NER example. Source: Mehmood et al. [94].

Relation extraction (RE) is an IE task consisting of identifying and labeling relations described in the text between named entities [130]. Relations can occur between any number of named entities scattered throughout the text [129]. Some common relations extracted from clinical text are associations between a medical condition and its anatomical location (condition-location) [34], a medical condition and the administered treatment (condition-treatment) [152], a medical procedure and its anatomical location (procedure-location) [55], and interactions between drugs (drug-drug) [130]. Figure 2.3 shows a clinical RE example where the relations between a drug and its adverse effects are identified.

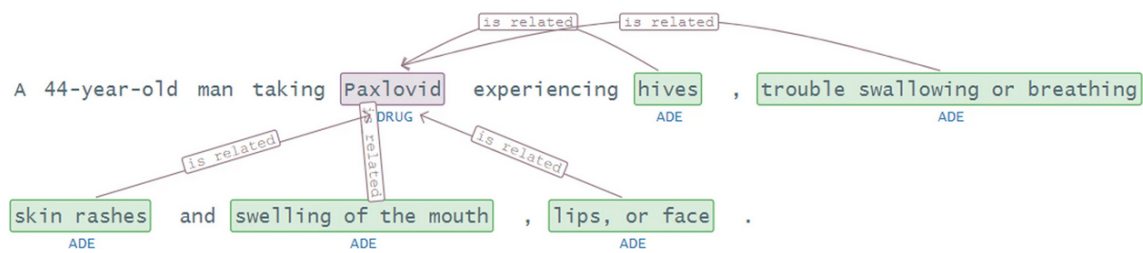


Figure 2.3: Clinical RE example showing drug-adverse effect relations. Source: Raza and Schwartz [115].

Entity linking (EL) or normalization is an IE task consisting of matching each recognized entity to the unique concept that defines it. The unique concept is in a target knowledge base and is identified by a unique code [36]. Linking medical terms in clinical text documents to knowledge bases connects the documents and terms to rich semantic information, which is essential for downstream applications [125]. Figure 2.4 displays the EL task applied to a clinical note, where three entities were identified and linked to their unique concept in a knowledge base.

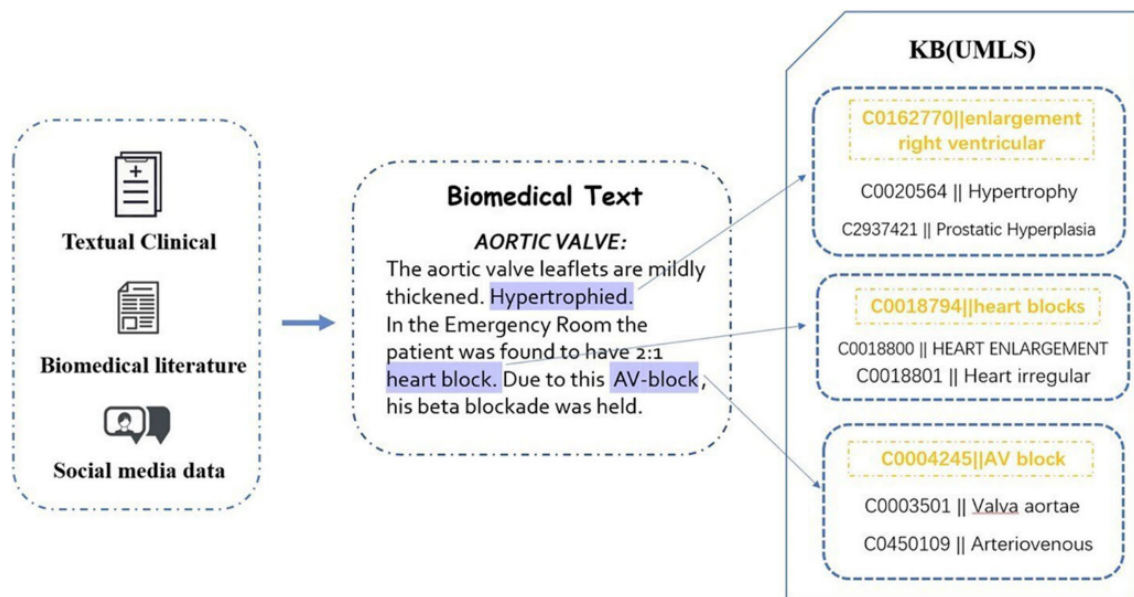


Figure 2.4: Clinical EL example. Source: Shi et al. [125].

Event extraction (EE) is an IE task consisting of identifying and extracting event expressions and their associated temporal information from unstructured text documents. In the clinical domain, this task involves extracting significant medical occurrences from clinical notes, such as diagnoses, treatments, and adverse events [82]. Figure 2.5 shows an example of event extraction on a clinical record, complete with temporal information.

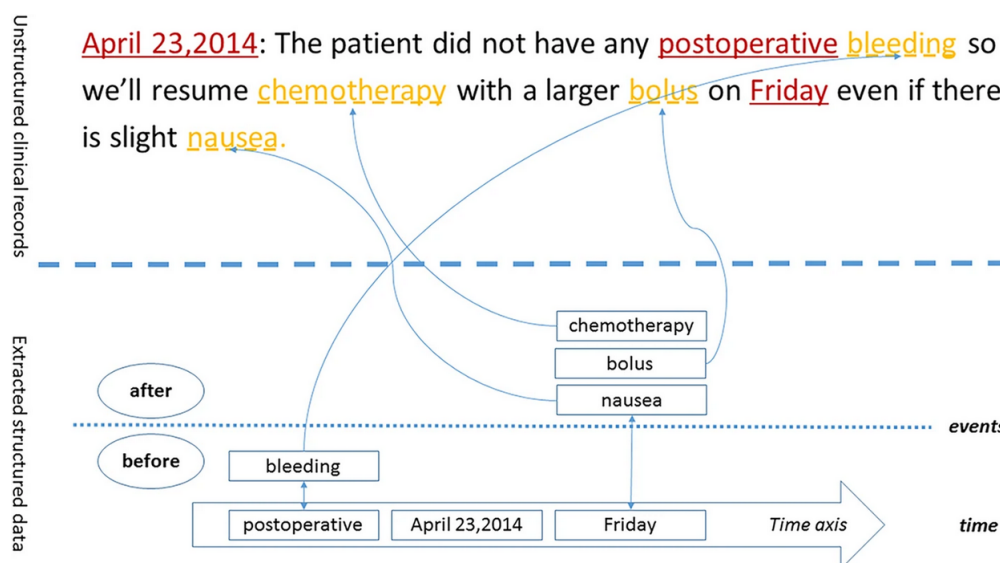


Figure 2.5: Clinical EE example. Source: Li et al. [82].

2.2.2 Applications

IE, as a task, is most useful in applications where the volume of textual data to be analysed is unfeasible for human examination and semantic information appears at a high frequency [47]. It serves as a foundation for various downstream applications, such as knowledge reasoning, knowledge graph construction, and question answering [157]. Due to recent advancements in AI, especially with regard to large language models and their text understanding capabilities, IE has garnered significant attention [157].

IE has various different applications in the clinical domain, such as improving patient care, supporting clinical decision-making, enabling large-scale epidemiological and pharmacological studies, supporting patient selection for clinical trials, and optimizing clinical workflows [150]. Liu et al. [84] developed a large language model specialized in extracting eligibility criteria entities from clinical trial documents. Raza and Schwartz [115] created an approach to extract clinical and demographic entities, as well as their relations, from COVID-19 case reports. Sciannameo et al. [123] evaluated a large language model's ability to extract clinical information from medical case reports, focusing on identifying key patient details such as sex, age, condition, and treatment.

2.2.3 Challenges

While IE offers general challenges, clinical IE poses various domain-specific, significant challenges, mainly due to characteristics inherent to clinical text [166]. Clinical text contains ambiguity, where distinct medical concepts can be mentioned through very similar (or even equal) forms [169], especially in the case of acronyms [166]. For example, 'depression' can mean the medical disorder or a drop in a biological parameter [36]; the acronym 'ER' can be the estrogen receptor gene or the endoplasmic reticulum [166]. Furthermore, the same concept can have different names [125, 36]: 'lung cancer' can be referred to as 'lung neoplasm malignant' [18]. It is

also abundant in complex, domain-specific terminologies [169], morphological variations, word order variations, and synonyms [125]. Another prevalent feature of clinical text is informality: typos, abbreviations, and the use of colloquial terms often found in clinical text may lead to mentions differing significantly from their corresponding entities in a knowledge base. For example, a healthcare specialist may write "renal disorder" when referring to "nephrosis" [149]. The heterogeneity of text sources also poses a challenge, as clinical text comes from various sources and may be characterized by a particular style of a physician, nurse, or institution, causing discrepancies [149].

Lastly, IE and other NLP tasks tend to require large amounts of annotated data to train, validate, and test models, especially in the case of recent deep learning models [125]. Obtaining a useful amount of annotated *corpora* for the clinical domain poses a difficulty, due to the need to de-identify the data and the specialization required to annotate it [125]. One of the biggest consequences of this difficulty is that languages that are not English have few annotated *corpora* and, thus, have fallen behind in terms of research and development [120].

2.3 Neural networks

Neural networks (NN) are a parallel combination of small processing units, called *nodes*, that obtain knowledge through a learning process and store that knowledge in the connections between the nodes, called *edges* [130]. They are represented by directed graphs [49] and are designed to model the way the human brain functions, where *neurons* (nodes) are connected by *synapses* (edges) [57]. Furthermore, NNs use fully connected layers, meaning every node in a layer is connected to every node in another layer [130].

Neural networks can learn high-quality distributed vector representations for text spans, called embeddings. These representations capture the syntactic and semantic features of the text, grouping similar text spans together [99]. The embeddings are used by NN-based models to perform NLP tasks. Embeddings propelled NN-based models to state-of-the-art results in NLP tasks such as IE [36, 125].

2.3.1 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a type of neural network consisting of three parts: the input layer, the output layer, and the hidden layers. The hidden layers typically consist of convolutional layers, pooling layers, fully connected layers, and normalization layers [159].

A CNN receives as input a text span represented as a sequence of vectors. In the convolutional layers, a weight matrix, called *kernel* or *filter*, slides across the sequence computing the dot product with the vectors. The resulting matrix of dot products, called a *convolved feature* or *feature map*, is passed to the pooling layers. The convolution operation reduces the number of free parameters, allowing the CNN to go deeper with fewer parameters [159]. The pooling layers slide a kernel across the convolved feature that applies an aggregation function, producing an output matrix that is smaller than the original input, therefore reducing the output dimensionality while keeping the

most notable information [159]. Subsequently, a fully connected layer performs classification on the output of the pooling layers, usually through a softmax function [83]. Normalization layers appear throughout the network to standardize the data, in order to improve the network's learning rates. Figure 2.6 depicts a generic CNN-based model for an NLP task.

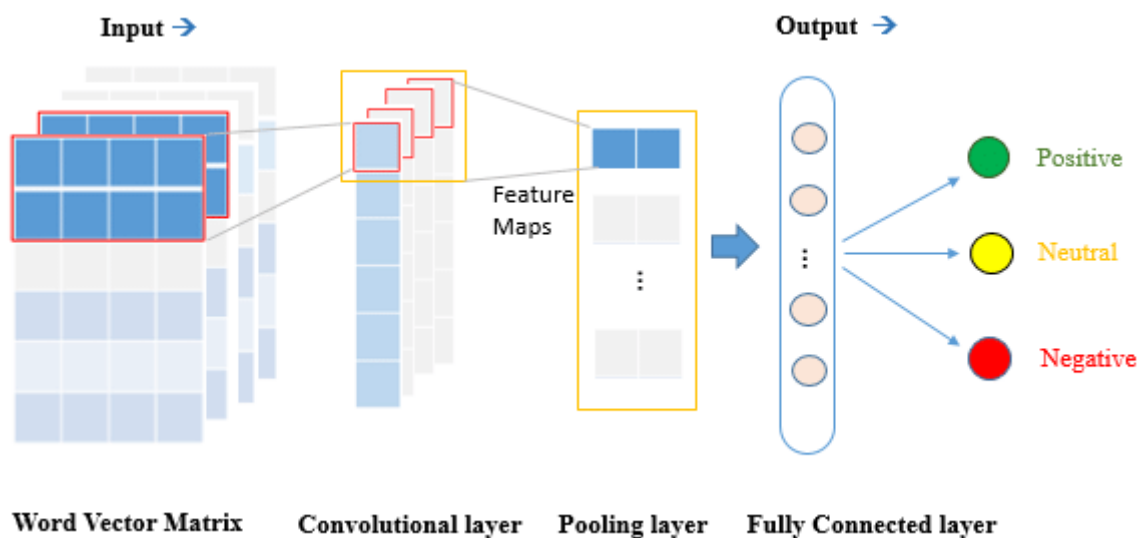


Figure 2.6: CNN-based model for NLP. Source: Tam et al. [135].

2.3.2 Recurrent Neural Networks

A Recurrent Neural Network (RNN) is a type of neural network that distinguishes itself by its ability to maintain a memory of previous input in its internal state. Much like other neural networks, RNNs consist of an input layer, an output layer, and hidden layers. However, the hidden layers in an RNN are composed of nodes that hold an internal state, or *memory* [130]. RNNs work by passing the sequential data that they receive to the hidden layers one step at a time. Each time a node in the hidden layers receives an input, its internal state is updated. The output provided by the node is a function of the current input and the node's current internal state. This means RNNs are well-suited for tasks where the order and context of the input sequence are important, such as most NLP tasks.

Bidirectional Recurrent Neural Networks (BRNNs) are a variant architecture of RNNs where the output of the hidden layers also takes into account future elements in the sequence, as opposed to only previous ones [159].

Long Short-Term Memory (LSTM) networks are a variant architecture of RNNs proposed by Hochreiter and Schmidhuber [59] meant to outperform RNNs in handling long dependencies, that is, long text spans where words that are related might be far from each other [130]. RNNs struggle to deal with long dependencies because of the vanishing gradient problem, where the effect of a given input on the internal state of the nodes diminishes to the point of insignificance as the network moves further through the sequence. LSTMs address this through their units,

called *cells* [159]. A cell is an LSTM unit composed of three multiplicative gates - *input*, *output*, and *forget gates* - that control the proportions of information to forget and pass on to the next step [91]. This mechanism has, indeed, proved to be better at learning long dependencies than regular RNNs [88, 134]. LSTMs are also combined with BRNNs to form Bidirectional-LSTMs (BiLSTMs), which use two layers of cells to produce an output, one for the previous and another for the future text in the sequence [130]. This approach has proved useful [154]. Gated Recurrent Units (GRUs) [19] are similar to LSTM units. They are composed of two multiplicative gates: the *reset gate*, which is analogous to the forget gate in LSTM, and the *update gate*, which can be considered as a combination of the input and output gates [143].

2.3.3 Transformers

Transformers were introduced in 2017 by Vaswani et al. [147]. Transformer-based systems have become the foundation for most state-of-the-art clinical IE approaches [36]. Transformers have an encoder-decoder architecture. The encoder takes an input sequence and processes it. Attention mechanisms capture relationships and allow each element to consider dependencies across the entire sequence. Positional encoding addresses sequence order. With this, the encoder outputs a sequence of embeddings that capture the contextual information of each input token [130]. The decoder, on the other hand, takes the encoder's output and generates an output sequence autoregressively, one token at a time, attending to the encoder's representations [130] and using similar mechanisms to those of the encoder. This architecture means Transformers are good at capturing long-range dependencies and processing sequential data [130]. Figure 2.7 summarizes the Transformer architecture. The most influential Transformer-based systems in clinical NLP tasks are large language models, namely generative models and Bidirectional Encoder Representations from Transformers (BERT).

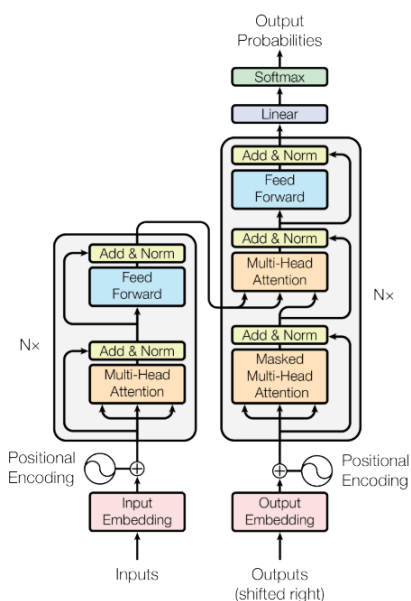


Figure 2.7: Transformer architecture. Source: Vaswani et al. [147].

2.3.4 Embeddings

In the context of NLP, embeddings are representations of text as vectors in a vector space, consisting of fixed-sized numerical vectors that aim to capture syntactic and semantic relationships between spans of text [130]. Systems learn embeddings through unsupervised pre-training using large data sources [98, 130]. Initially, embeddings were static representations of text spans [36], meaning a word or sentence would always be represented by the same vector, regardless of context. For example, in the sentence "I saw a bat flying around while I was carrying my baseball bat.", static embeddings would represent both instances of 'bat' in the same way, even though they have very different meanings. This was the case with the embeddings of Word2Vec [98] and GloVe (Global Vectors for Word Representation) [110], for instance. Contextual embeddings greatly improved the performance of many NLP tasks [36]. Contextual embeddings are powerful due to the fact that their representations of text take context into account. The same word or sentence in two different contexts generates two different embeddings [130]. ELMo [122] and BERT [27] are two prime examples of models that generate contextual embeddings of text [36, 130].

2.4 Language Models

Language models (LMs) are computational models capable of understanding and generating natural language. LMs are able to predict the likelihood of word sequences or generate new text based on a given input [16].

2.4.1 Large Language Models

Large language models (LLMs) are advanced language models based on neural networks, with massive parameter sizes and pre-trained on large amounts of data. They can perform a wide variety of tasks, even many not related to natural language [5, 16, 100]. LLMs are more complex than other ML models. They also have more parameters and training data [100]. Table 2.1 denotes the differences in several criteria between traditional ML models, DL models, and LLMs.

Comparison	Traditional ML	Deep Learning	LLMs
Training Data Size	Large	Large	Very large
Feature Engineering	Manual	Automatic	Automatic
Model Complexity	Limited	Complex	Very Complex
Interpretability	Good	Poor	Poorer
Performance	Moderate	High	Highest
Hardware Requirements	Low	High	Very High

Table 2.1: Comparison between LLMs and other types of machine learning models. Source: Chang et al. [16].

Since the release of ChatGPT in late 2022 [2], LLMs have garnered significant attention due to their strong performance on a wide range of natural language and domain-specific tasks [16, 100].

Some of the current main LLMs are BERT [27], GPT-3 [33], GPT-4 [105], LLaMA [141], and InstructGPT [106].

LLMs use Transformer architectures [16, 100, 51], explained in Section 2.3.3. The following key features make these models outperform other ML models in learning and understanding the complex patterns and structures present in language data [51]: in-context learning, through which LLMs are able to learn how to solve a new task by being provided a small set of examples during prompting [100]. This means they are better than other ML models at providing contextually relevant solutions and answers [16]; Reinforcement Learning from Human Feedback (RLHF), through which LLMs can be continually fine-tuned using feedback collected from interactions with users and environment, thereby improving their capabilities over time [16, 100]; instruction following, through which LLMs are able to follow instructions for new tasks without receiving explicit examples [100]. This is done through *instruction tuning*, where an LLM is fine-tuned using various multi-task datasets formatted via natural language descriptions [167]; multi-step reasoning, through which LLMs are able to solve a complex task by breaking it down into intermediate reasoning steps [100]; and transfer learning, where the knowledge that LLMs acquire during their pre-training on massive amounts of text can be leveraged for downstream tasks without the need for retraining the model from scratch, which would be computationally expensive and require large labeled task-specific datasets. Instead of training a separate model for each specific task, a pre-trained LLM can be fine-tuned on smaller task-specific datasets. Therefore, through transfer learning, the development of effective language models requires far less computational power and data [52]. Figure 2.8 provides an overview of an LLM’s capabilities.

LLMs have also been shown to perform well on domain-specific tasks, and have been used in domains such as healthcare, finance, law, education, and others [167].

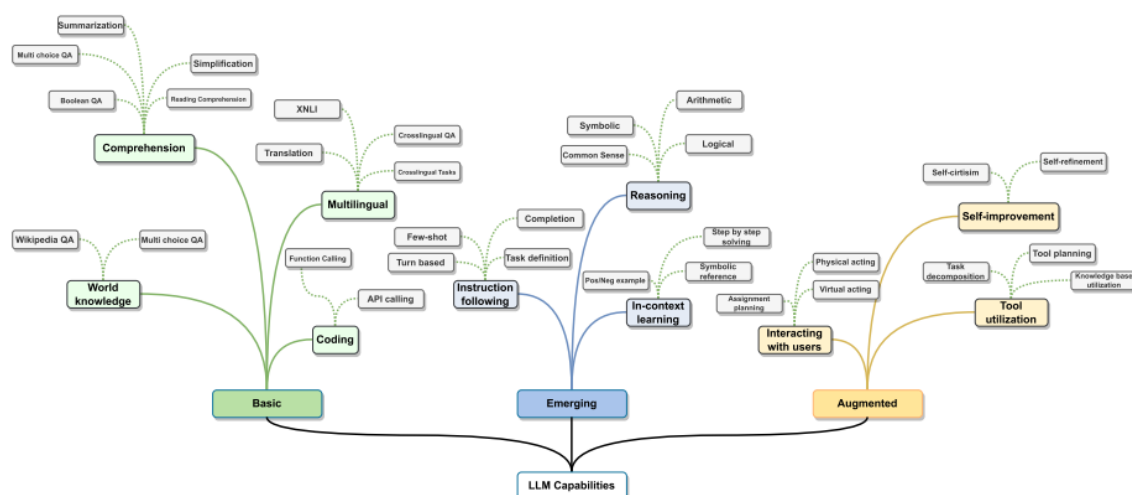


Figure 2.8: LLM capabilities. Source: Minaee et al. [100].

2.4.2 Generative large language models

Generative LLMs are large language models capable of generating new, original content that is not explicitly programmed [162]. They learn the properties and patterns of the training data and use them to create new samples that resemble the training data [52]. They can be aimed at different tasks, usually through task-specific fine-tuning. The tasks include text inference, whereby generative LLMs are capable of understanding and extracting meaning from natural language. Through this, they are also able to generate text, predict next words, summarize and translate text, make logical assumptions, maintain a regular conversation, among other abilities [52]; image generation and analysis, by which they can be used to generate images and can also understand and analyse images provided to them [50]; instruction following, whereby generative LLMs have the ability to follow instructions, given as if they were being layed out to a human, and perform tasks based on those instructions. They can also make use of provided examples to better understand and complete the requested task [16, 52]. This task is responsible for the increase in attention in the eyes of the general public, mainly through ChatGPT [100]; code generation, where they have shown they are capable of generating code from scratch and completing code provided, thus improving programming productivity by reducing time spent writing code [52]; and others [52, 100].

Some of the state-of-the-art families of generative large language models include OpenAI's GPT (GPT-2 [113], GPT-3 [33], GPT-3.5 [105], GPT-4 [106]), Meta's Llama (Llama [141], Llama 2 [142], Llama 3 [140]), and Google's Gemini [138].

2.4.3 BERT

BERT [27] is a bidirectional embedding model with a Transformer-based architecture. BERT represents input and output sequences using deep contextual embeddings [36]. Each sequence corresponds to a pair of 'sentences', where a sentence is a span of contiguous text. Sequences are tokenized using wordpiece embeddings [156] and the final representation of a token is a combination of the token itself, its position, and the sentence it belongs to. This contextual, double-sentence embedding means BERT is good at capturing contextualized meaning and handling long dependencies. Figure 2.9 depicts an example of BERT's deep contextual embeddings.

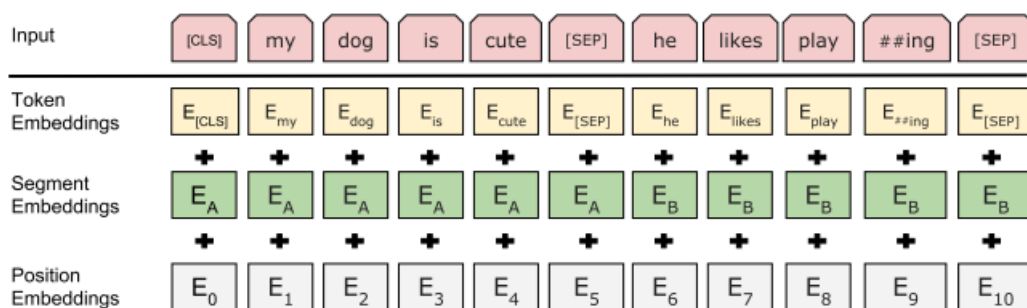


Figure 2.9: An example of a BERT embedding. Source: Devlin et al. [27].

BERT's framework consists of two steps: pre-training and fine-tuning. During pre-training, the model is trained on unlabeled data over two different pre-training tasks. One is a Cloze task [137], which basically means the model is deeply bidirectional, as opposed to the other systems [122], which are just a shallow concatenation of independent left and right context. The other task is a Next Sentence Prediction (NSP) task so the model can learn to understand relationships between two sentences. When being used for a downstream task, a BERT model is initialized with a set of pre-trained parameters and these are fine-tuned using task-specific labeled data from the target downstream task. The same pre-trained parameters can be easily fine-tuned for multiple different downstream tasks. Figure 2.10 shows the overall pre-training and fine-tuning procedures for BERT.

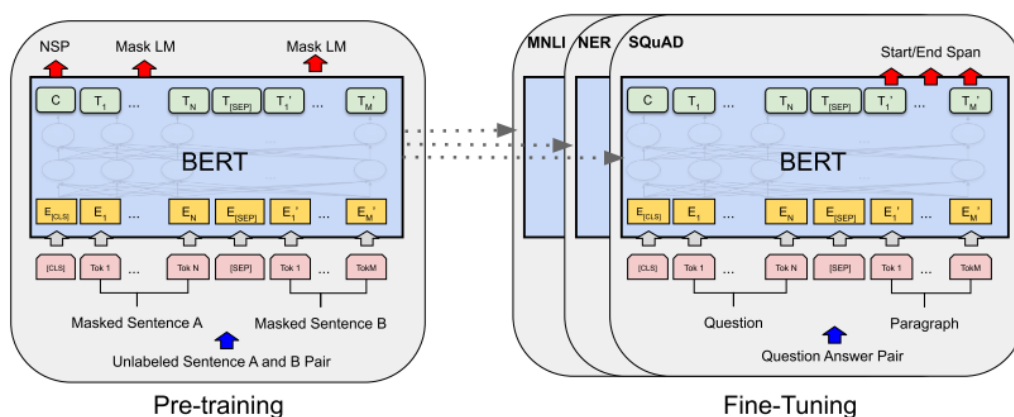


Figure 2.10: Overall pre-training and fine-tuning procedures for BERT. Source: Devlin et al. [27].

BERT has been widely used as a basis for systems that perform IE and other NLP tasks [36, 65, 125]. Some BERT-based systems that performed well in their respective NLP tasks are BioBERT [80], ClinicalBERT [62], Multilingual BERT [27], and BERTimbau [132].

2.4.4 LLM-based information extraction

Transformer-based models, such as LLMs, have greatly improved the state-of-the-art for IE and other NLP tasks, as they allow the modeling of complex contextual relationships within text [71]. Since IE is a sequence labeling problem and not a text generation problem, generative models have shown limitations, and domain-specific BERT-based models generally perform better [61, 71]. However, BERT-based models are supervised and producing high-quality training data for clinical and other domains is costly. Therefore, the ability of generative models to perform the task with no training data and few examples is very attractive [70].

The process of developing a BERT-based model to perform IE is similar across domains [15, 62, 80, 132]. The original BERT model was pre-trained on the English Wikipedia and Book-Corpus, a large *corpus* of self-published books [27]. Domain-specific models generally use the original BERT weights and perform further pre-training on large *corpora* of domain-specific data.

This is important for the clinical domain, as it is complex and specialized, with a considerable amount of domain-specific terms: IE models designed for general-purpose language understanding often obtain poor results in clinical text mining tasks [80]. Subsequently, models are fine-tuned using datasets labeled with the intended IE task, so the model can learn to perform it, specifically [62, 80].

Generative models were originally designed for text generation tasks, causing them to struggle with IE. To address this, various methods have emerged [71]. These typically involve *prompt engineering*, where prompts are customized to better fit the intended IE task, by providing specific annotation guidelines, examples, entity definitions, among other strategies [70]. Another common method, which can be used in conjunction with *prompt engineering*, is *instruction tuning*, which leads to improved zero-shot performance on IE [70]. However, this method requires high-quality domain-specific data, making it costly and eliminating the cost advantage of generative models over BERT.

Chapter 3

Synthetic clinical data generation

In this chapter, we begin by defining synthetic data generation and providing an overview of its methods, along with a summary of clinical studies that used them. Then, we explain how synthetic data has the potential to have advantages over real data in scientific applications. Afterward, we review the state-of-the-art in LLM-based generation of synthetic unstructured clinical textual data. In this review, we detail techniques seen in literature for data pre-processing, model fine-tuning, prompt engineering, and synthetic data evaluation. Furthermore, we describe their use in various studies. Lastly, we discuss challenges commonly associated with the task and end the chapter by providing some conclusions about synthetic clinical data generation.

3.1 Synthetic data generation

Synthetic data is computer-generated data used to augment or replace real data in tasks such as training AI models [6]. Synthetic data encompasses various types, including structured tabular data, unstructured text, images, audio, and video, each generated using specific methods tailored to the respective data type [10].

Synthetic datasets can be a good alternative to overcome the problem of obtaining large amounts of real data for applications that require them. If generated correctly, they present other benefits, such as reducing biases [77, 144], boosting performance, being scalable, and maintaining privacy [69]. Synthetic data has been used in many different fields, such as human analysis [69], medical imaging [40], finance [54], and others [17].

3.1.1 Overview of methods

Various types of methods have been used to generate synthetic data, from rule-based methods to statistical and deep-learning methods [10].

Rule-based methods involve creating synthetic data by applying predefined rules, logical constraints, and domain-specific knowledge. This deterministic approach ensures that generated data follows specific patterns and relationships observed in the real world, making it particularly useful for structured data applications like software testing [46].

Statistical methods employ mathematical and probabilistic techniques to replicate the statistical properties of real datasets. These methods are particularly effective for structured data, such as tabular datasets, and typically have the advantage of interpretability and lower computational requirements [46]. Pivotal statistical approaches include parametric and non-parametric methods and Bayesian networks. In parametric methods, models assume that the data follows a specific statistical distribution and estimate the parameters of this distribution from the real data. Generated synthetic data follows the estimated parameters [10]. In non-parametric methods, models do not estimate parameters of data distribution: the distribution is modeled directly from the data, allowing for more flexibility in capturing complex data structures [112]. Bayesian networks are models that represent variables and their conditional dependencies using a directed acyclic graph. Among statistical methods, they are particularly useful for generating synthetic data that preserves relationships and dependencies [161].

Deep learning methods have revolutionized synthetic data generation due to their ability to learn complex, high-dimensional data distributions, which enables the creation of realistic synthetic datasets [10, 56]. These methods include generative adversarial networks (GANs), variational autoencoders (VAEs), diffusion models, and generative LLMs.

GANs [45] are a type of deep learning model designed to generate synthetic data by setting up a competitive framework between two NNs: a generator and a discriminator. The generator aims to create synthetic data that mimics the real data, while the discriminator attempts to distinguish between real and synthetic data. Through this process, the generator progressively improves its output until the discriminator can no longer consistently tell the difference.

VAEs [75] are probabilistic generative models that work by learning a compressed latent space representation of the input data. A VAE encodes input data into a lower-dimensional latent space, imposing a probabilistic data distribution, and then decodes samples from that space to reconstruct the data. This process allows the model to recreate the input data and generate new data samples by sampling from the latent distribution.

Diffusion models are generative models that work by learning how to reverse a gradual noise-adding process, called diffusion. During training, data samples are progressively corrupted by adding small amounts of noise over many steps until they resemble pure noise. The model then learns to reverse this process until it is able to reconstruct a sensible data sample from noise [168].

Throughout the years, generation of synthetic clinical data has made use of the methods described in this section. To highlight the diversity in methods and clinical data types seen in literature, Table 3.1 summarizes some examples of synthetic clinical data generation studies.

Study	Description	Data type	Method(s)
Kharya et al. [72]	Generation of structured patient records of heart disease, breast cancer, and diabetes patients	Tabular	Bayesian networks
Shi et al. [124]	Generation of high-fidelity, privacy-conscious structured hypertension patient records for causal effect estimation with multiple treatments	Tabular	GAN
Yao et al. [160]	VAE pre-training using blood cells and breast cancer tissue imaging datasets artificially balanced with synthetic random oversampling to mitigate bias against minority classes	Imaging	VAE, GAN
Han et al. [53]	Contrastive diffusion model with auxiliary guidance for coarse-to-fine positron emission tomography scan reconstruction	Imaging	Diffusion model
Dash et al. [24]	Exploration of different GAN types to produce synthetic longitudinal patient profiles	Time-series	GANs
Khorchani et al. [73]	Generation of synthetic longitudinal observational patient cohorts from COVID-19 clinical data	Time-series	Parametric model
Tang et al. [136]	Generation of synthetic, labeled clinical literature-style sentences to be used in fine-tuning models for downstream tasks	Unstructured text	LLM

Table 3.1: Studies on generation of synthetic clinical data of various types.

3.1.2 Applications

The goal of synthetic data is to replace and supplement real data in circumstances where the latter is difficult or costly to obtain. Therefore, generated synthetic data has the same applications as real data [77]. The most prevalent recent application of synthetic data is training LLMs, which need high quantities of general-domain and domain-specific data. Often, real data is unable to fulfill this necessity [85]. However, synthetic data has the potential to bridge this gap and also improve and solve various limitations of using real data, some of which are very important in the clinical domain [145].

Many real datasets contain sensitive personal information. Therefore, sharing these datasets for research is limited by privacy concerns, which hinders development of research areas that need them. For such datasets to be used, they require anonymization and de-identification procedures performed by specialists, which are costly in terms of time and human resources. Synthetic data can address these concerns. By training generative models with private real data, generated synthetic samples cannot be linked to an individual or reveal private information [145]. This is especially important in the clinical domain, as real clinical data contains plenty of sensitive personal information that can be easily traced back to a specific individual [145].

Another issue that synthetic data can address is bias. AI models reflect the bias of their training data, which can cause issues in their employment in downstream applications. Recently, the challenge of bias has increased in importance, as large language models are trained on vast amounts of real data that inevitably contain biases [145]. Through the manual enforcement of fairness constraints and requirements, we can use generative models to generate fair and unbiased synthetic datasets. These would improve the quality and fairness of downstream applications that use LLMs trained on these datasets [145]. In the clinical domain, generating diverse synthetic data that includes underrepresented groups or rare conditions may help train models that are more generalizable and equitable, allowing minority characteristics to be properly taken into account [101, 145].

Synthetic data can also assist in dealing with insufficiency of domain-specific data. Datasets of data related to the target domain can be leveraged by generative models to generate synthetic data that has adaptations toward the intended domain [145]. Another related ability is simulation, whereby synthetic data can be used to simulate hypothetical situations or real situations for which there is no available data. Simulations can be heavily controlled, facilitating hypothesis testing, guiding experimental design, and accelerating scientific discovery [145]. In the clinical domain, these features can enable the exploration of rare disease scenarios, evaluation of treatment outcomes under diverse patient conditions, and development of diagnostic tools. By simulating patient trajectories and interventions, synthetic data supports personalized medicine research and improves clinical decision-making models [145].

3.2 LLM-based generation of synthetic unstructured clinical textual data

The NLP task-solving improvements brought on by the advent of large language models reinforced the fact that large amounts of high-quality data are key to building robust NLP models [85]. As explained in [Section 3.1](#), synthetic data is seen as a potential alternative to human-generated data. Due to their pre-training size, LLMs have obtained a great deal of knowledge that gives them great natural language comprehension [28]. This comprehension equips generative LLMs with a remarkable ability to generate unstructured textual data [158]. Several studies have exploited this ability and generated clinical data. Since the goal of this project is to generate a dataset of synthetic clinical notes, which are unstructured textual documents present in electronic health

records, in this section we discuss those studies and paint a picture of the generic generation process of unstructured clinical textual data.

Electronic health record (EHR) is the longitudinal collection of electronic health information about a patient, collected during routine delivery of healthcare [23, 48]. An EHR stores clinically relevant patient-level information, such as demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data and radiology reports [1]. This information is collected and maintained over time by the patient’s various healthcare providers. A clinical note is an unstructured textual document that registers a component of care provided to a patient. Examples are discharge summaries, admission notes, and ambulatory notes [104]. These notes contain detailed information on the patient’s status, care provided, and notable occurrences during the circumstance at hand. Clinical notes are a common type of document in EHRs [25].

Studies covered in this section were collected through searches in Google Scholar¹, ACM Digital Library², and IEEE Xplore³. Our searches were directed at studies that generated datasets of synthetic unstructured clinical textual data using LLMs. In studies matching this criteria, we analysed the process through which their datasets were generated.

As a result of this analysis, in this exposition of the process of generating synthetic unstructured clinical textual data, we have elected to divide it into three parts: *preparation*, *prompt engineering*, and *evaluation*. For each specific part of the process, we describe the methodology followed by the studies that explicitly described that part in their articles.

3.2.1 Preparation

This stage involves data pre-processing and model fine-tuning, if performed. Various studies opt to rely on the properties the LLM learned from its pre-training data [30, 136], coupled with using some examples [35, 89, 158]. However, some attempt to enhance the model’s capabilities by fine-tuning it on task-specific data.

Peng et al. [109] collected 290 million clinical notes from the UF Health Integrated Data Repository and 195 billion English words from a high-quality *corpus*, merging the two *corpora* after de-identifying the clinical notes. Afterward, they normalized the *corpus* by unifying all text into UTF-8 encoding, removing illegal UTF-8 strings, removing HTML/XML tags, and tokenizing the text, followed by other minor adjustments. They utilized this data to train their GPT-based LLM, making it more viable for clinical NLP tasks.

Amin-Nejad et al. [9] began by extracting 55,404 EHRs from the publicly available MIMIC-III *corpus* [68]. They tokenized the text, converted it to lowercase, removed tokens that occurred fewer than 4 times, and replaced newline characters with a special token. They used this processed *corpus* to fine-tune a smaller version of the GPT-2 model.

Kumichev et al. [76] fine-tuned an LLM on two datasets: a Russian instruction-following medical dataset, to improve the model’s medical knowledge and its ability to generate Russian

¹<https://scholar.google.com>

²<https://dl.acm.org/>

³<https://ieeexplore.ieee.org/Xplore/home.jsp>

text based on instructions, and a Russian language dataset, to further improve the model’s ability to generate Russian text. The first dataset was composed of 89% real medical data from various sources and 11% synthetic clinical notes generated by ChatGPT. The second dataset was composed of instructions and dialogues generated by GPT-4. Data pre-processing is not described.

As we can see, the existence and details of a preparation stage are dependent on the case at hand, and no general practice can be inferred.

3.2.2 Prompt engineering

In order to create the synthetic data, the model must be specifically prompted to do so. Developing and optimizing ways to prompt an LLM to reach a desired result is a task called *prompt engineering* [43]. The LLM must be informed of the purpose of the task, what the required data entails, its format, and any other knowledge necessary in understanding the task, similarly to what would be done with a human [42]. There are various examples of widely different approaches to *prompt engineering* in the clinical domain. Table 3.2 summarizes the prompt engineering strategies of the collected studies, which are then further detailed. The first four studies discussed in this section disclosed the final prompt version used to generate their respective datasets. We list these prompts in Appendix A.

Study	LLM	Prompt engineering strategy	Generated data
Falis et al. [30]	GPT-3.5	Eight points of specification to tune generation. Zero-shot prompting	Discharge summaries with ICD-10 codes for entities mentioned
Tang et al. [136]	ChatGPT	Three cycles of five model-suggested, manually-tested prompts, to arrive at final prompt. Zero-shot prompting	PubMed-like, entity-specific sentences
Lorge et al. [89]	ChatGPT	Exploratory prompt optimization. Used error-correcting examples (few-shot prompting) and conditional prompting for diversity	1,000 clinical notes, annotated with entities
Xu et al. [158]	ChatGPT	Extraction of disease concepts from knowledge base. Request for ChatGPT to list entities of specific types. Infusion of this knowledge into prompts structured with instructions and examples (few-shot prompting)	18 datasets of clinical notes tailored for 8 clinical NLP tasks
Frei and Kramer [35]	GPT-based	Prompt to replicate twelve annotated examples encoded in markup-like style (few-shot prompting)	Clinical sentences labeled with drugs, dosage, and diagnosis
Latif and Kim [79]	ChatGPT	Augmentation of real dataset of clinical sentences by requesting model to rephrase them. Zero-shot prompting	Not specified

Table 3.2: Prompt engineering approaches used by various studies.

Falis et al. [30] required prompts with eight points of specification when applying GPT-3.5 to generate a dataset of discharge summaries containing ICD-10 codes for the concepts mentioned.

Tang et al. [136] attempted to leverage ChatGPT’s self-understanding by prompting it to suggest ‘five concise prompts or templates that can be used to generate data samples of [Task Descriptions].’, generating 10 samples using each suggestion, and professionally evaluating them to pick the best prompt. Afterward, they again prompted ChatGPT to suggest five more prompts using the best prompt from the previous round (manually tested with 10 data samples), repeating this process three times. The final prompt was used to generate sets of 30 sentences containing a specific medical entity mimicking PubMed articles.

Lorge et al. [89] used ChatGPT to generate and annotate at the NER level a dataset of 1,000 clinical notes. They empirically optimized their prompts and reached a resulting prompt that began with ‘Create a narrative psychiatric clinical note (about 500 words) and annotate sentences with[...]’, followed by a description and example for each of the labels from their annotation schema. They found that the best prompts balance providing examples that specifically aim to correct recurrent mistakes in the model’s behaviour against the tendency for the model to often repeat provided examples. Furthermore, they observed that it was necessary to request the model to provide notes with specific labels in order to achieve enough diversity - a process called *conditional prompting*.

Xu et al. [158] designed a detailed framework for LLM-based synthetic clinical data generation, called ClinGen. Figure 3.1 gives an overview of their prompt engineering approach. They start by injecting information from an available high-quality knowledge base, through the extraction of its disease concepts. They also leverage the fact that ChatGPT contains medical information as training data and prompt it to generate concepts of certain types: ‘Suppose you are a clinician and want to collect a set of <Entity Type>. Could you list 300 entities about <Entity Type>?’. Subsequently, they inject these medical concepts into the prompts, as shown in Figure 3.1. They employ this framework to generate 18 datasets suited for 8 different clinical NLP tasks.

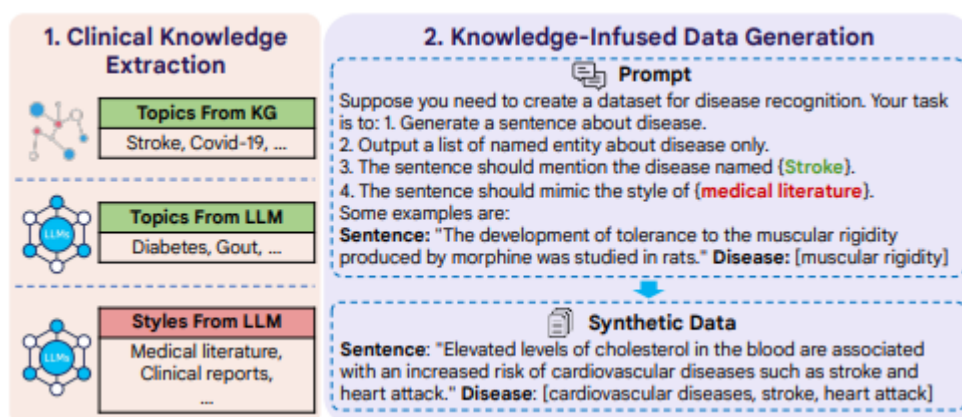


Figure 3.1: ClinGen prompt engineering approach overview. Source: Xu et al. [158].

Frei and Kramer [35] prompted a GPT-based model with an input sequence of twelve sentences in German (few-shot prompting), encoded in a pre-defined markup style and annotated with the

label classes *drug*, *dosage*, and *diagnosis*. They asked the model to generate more sequences with the same format as the twelve examples and removed results according to some filters, such as duplicates.

Latif and Kim [79] used ChatGPT to augment a real dataset of clinical sentences by taking a real sample and prompting the model to ‘rephrase the following sentence for prevention, treatment, or risk factor after the word because/since/as’, without using examples.

3.2.3 Evaluation

Evaluation of synthetic data is independent of the generation method used. Therefore, the goals and methods described in this section refer to any synthetic clinical data, not just data generated by LLM-based techniques.

Properly evaluating synthetic data is crucial in determining whether it can effectively replace real data in NLP tasks. There are many evaluation techniques for synthetic datasets, typically tailored to specific types of data or specific domains [29, 32]. However, there are several goals common to each synthetic dataset, with the weight of each criteria varying according to domain [29]. In the clinical domain, we can state that the objective is generating useful, faithful, and diverse data that cannot be traced to any real individual [145]. Thus, the desired dimensions of a synthetic clinical dataset are *utility*, *fidelity*, *diversity*, and *privacy*. Utility means the data must be useful enough to replace real data for predictive and modelling purposes [14]. Fidelity means the synthetic samples must mimic real samples, generally those from which the generative model learned ground truths, thus being correct, coherent, and relevant [14, 32, 85]. Diversity means the data must replicate the natural variations in human-generated data, generating many unique data-points [29, 85]. Privacy means synthetic datasets must ensure their data does not represent any concrete individuals [29].

Generally speaking, privacy and the other three are inversely related. It is simple to avoid privacy issues by not involving any real data in the process, but that tends to lower the quality of the dataset. Conversely, one can increase the quality by heavily making use of real samples, but that can easily make supposedly synthetic samples traceable to real individuals. Evaluation of datasets must take both sides into consideration [44].

Table 3.3 summarizes methods used to evaluate unstructured textual synthetic clinical data. ‘Method’ names the evaluation method, ‘Description’ gives its brief description, ‘Evaluated dimension(s)’ lists the dimensions the method assesses out of the four explained above, and ‘Used in’ lists clinical studies where the method was used. Below the table, each mentioned evaluation method is explained in detail. For methods that are not direct statistical calculations, we describe how they were employed in each of the listed studies.

Method	Description	Evaluated dimension(s)	Used in
Human evaluation	Medical specialists evaluate the quality of synthetic samples.	Utility, fidelity	[11, 20, 30, 95, 109, 153]
ML model performance	ML model is trained on synthetic samples, tested on real samples. Results are compared with model trained on real samples. May also train model on real samples and test on synthetic, or even train and test on synthetic only.	Utility, fidelity	[35, 109, 136, 158]
Perplexity	Measures uncertainty of AI model when predicting the next word in a sequence.	Utility, fidelity	[9, 77, 95]
t-SNE plot	Technique used for visualizing high-dimensional data in a lower-dimensional space. Converts pairwise similarities between data points into probability distributions.	Utility, fidelity, diversity	[136, 158]
Central Moment Discrepancy	LLM-specific method. Measures quantification of distribution shift of synthetic data in relation to real data.	Fidelity	[158]
Cosine similarity	Measures similarity between sentence-level embeddings of synthetic and real samples. For datasets, metric is used by calculating the similarity of all pairs of samples and averaging it.	Diversity, privacy	[95, 153, 158]
Hausdorff distance	Greatest of all distances from a point in one set to the closest point in another set. Thus, measures distance between subsets of a metric space.	Privacy	[58]
Textual statistics	Comparison of counting statistics between synthetic and real samples, such as entity and label counting statistics and document sizes	Utility, fidelity	[89, 158]

Table 3.3: Evaluation methods for synthetic clinical data.

Human evaluation

This class of methods consists of resorting to medical specialists to evaluate the quality of the synthetic samples. If the dataset is too large for complete human evaluation to be feasible,

one alternative is having a portion of the dataset analysed by humans and generalizing the conclusions [96, 151]. Multiple forms of human evaluation have been conducted.

Peng et al. [109] performed a Turing test of synthetic data. In this context, a Turing test consists of assessing whether synthetic samples are indistinguishable from real samples. They randomly sampled 30 real clinical notes and augmented them to form 30 synthetic notes. Then, they asked two physicians to try to determine which notes were real and synthetic.

Falis et al. [30] provided four clinical professionals with 10 real and 10 synthetic discharge summaries and requested their evaluation from 1 (failure to perform) to 5 (perfect performance) on utility and fidelity (under different names).

Choi et al. [20] had a medical doctor score how realistic 50 real and 50 synthetic clinical notes were, on a scale from 1 to 10 (10 being most realistic).

Beaulieu-Jones et al. [11] requested cardiologists to score from 1 to 10 real and synthetic blood pressure figures based on how realistically they depicted a patient.

Weldon et al. [153] generated patient EHRs and had medical professionals rank 20 real and 40 synthetic EHRs by how plausible they considered them to be as ICU patients, using a scale going from ‘highly implausible’ to ‘highly plausible’.

Melamud and Shivade [95] requested a clinician’s help in empirically analysing some glaring problems with the synthetic notes’ realism.

Human evaluation is very labour intensive but remains a crucial part of synthetic data evaluation. Thus, balancing human and automatic evaluation methods can help increase the rate of synthetic data generation, in an attempt to match the needs of the clinical field, while maintaining quality [85].

ML model performance

Given that one of the main purposes of synthetic data is training ML models [32], one method of evaluation is training a model on real samples and synthetic samples, separately, and testing both versions on a real test set. This is called ‘train on synthetic, test on real’ (TSTR) [29], as shown in Figure 3.2. Similar or better results are a strong indicator that the synthetic dataset is comparable to real data [158]. Conversely, another method of evaluation is training a model on real data and testing it on synthetic data - ‘train on real, test on synthetic’ (TRTS) [29]. In some cases, we can also find ‘train on synthetic, test on synthetic’ (TSTS) [35, 89].

Peng et al. [109] evaluated synthetic clinical data by comparing the performance of an LLM trained on that data with two other LLMs trained on real clinical data on five clinical NLP tasks: clinical concept extraction, medical relation extraction, semantic textual similarity, natural language inference, and question answering.

Tang et al. [136] attempted to create synthetic clinical data for the two IE tasks, NER and RE. They evaluated it by fine-tuning 3 language models with the synthetic data and the same 3 models with real clinical data, then comparing the precision, recall, and F1-scores achieved by both sets on high-quality datasets, on the two aforementioned tasks.

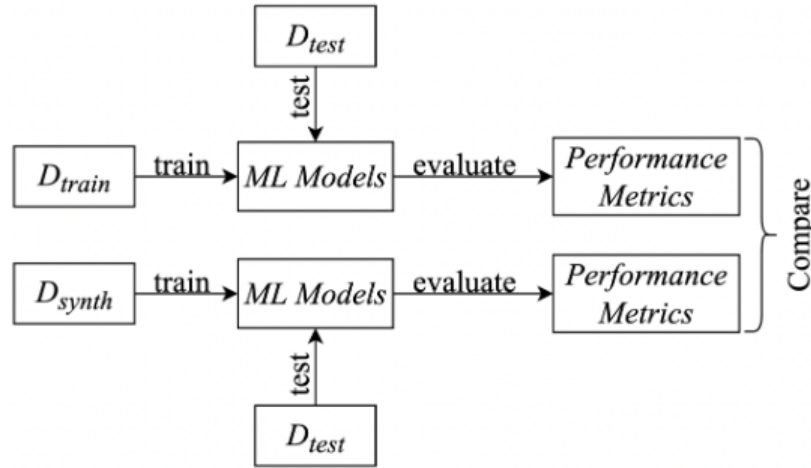


Figure 3.2: Diagram of ML model performance as an evaluation method. Source: Figueira and Vaz [32].

Xu et al. [158] compared 11 LLM-based synthetic data generation frameworks by generating 5,000 synthetic clinical notes using each of the frameworks and using them to fine-tune two biomedical BERT-based models. Afterward, they collected precision, recall, F1-score, and accuracy of all models on 18 high-quality datasets tailored for the following clinical NLP tasks: text classification, RE, natural language inference, fact verification, question answering, sentence similarity, NER, and attribute extraction.

Frei and Kramer [35] created a synthetic medical German dataset annotated at the NER level and evaluated the performance of 3 German BERT models by using the dataset to fine-tune and test the models on the NER task.

Perplexity

Perplexity (PPL) is a metric used to evaluate language models, which is defined as the exponentiated average negative log-likelihood of a sequence. If we have a tokenized sequence $X = (x_0, x_1, \dots, x_t)$ then the perplexity of X is

$$\text{PPL}(X) = \exp \left[-\frac{1}{t} \sum_{i=1}^t \log p_{\theta}(x_i | x_{<i}) \right] \quad (3.1)$$

where $p_{\theta}(x_i | x_{<i})$ is the log-likelihood of the i th token conditioned on the preceding tokens $x_{<i}$ according to our model [3]. Basically, it can be described as a model's uncertainty when predicting the next word in a sequence. Given that language models can now achieve low perplexity scores on real data, this metric can be used as a measurement to quantitatively evaluate the similarity of a synthetic dataset to real ones, by comparing the perplexity of a model on the synthetic dataset with its perplexity on real datasets of the same type [77].

Kweon et al. [77] fine-tuned an LLM on a high-quality real clinical dataset and calculated the model's perplexity on 5 different datasets, including their created synthetic one, in order to

compare the results and draw conclusions on the similarity of the synthetic samples with real ones.

Amin-Nejad et al. [9] evaluated two synthetic clinical datasets by employing them in training a vanilla Transformer model and GPT-2 [113] and testing the models' negative perplexity on test sets of the synthetic datasets.

Melamud and Shivade [95] evaluated a synthetic medical dataset by comparing a LSTM's perplexity on the synthetic dataset with its perplexity on real general-domain and medical datasets.

t-SNE

t-distributed stochastic neighbor embedding (t-SNE) [146] is a non-linear dimensionality reduction technique used for visualizing high-dimensional data in a lower-dimensional space, typically 2D or 3D. It works by converting the pairwise similarities between data points into probability distributions — first in the high-dimensional space using a Gaussian distribution, and then in the lower-dimensional space using a Student's t-distribution. The algorithm then optimizes the lower-dimensional representation by minimizing the Kullback-Leibler divergence between these distributions through gradient descent. In synthetic data evaluation, the goal is to have similar t-SNE plots of the synthetic and real datasets [136, 158], as this demonstrates utility and fidelity on the synthetic dataset's part. Similar spacing between data points helps demonstrate that the synthetic dataset is as diverse as the real one [158]. Figure 3.3 depicts an example of this.

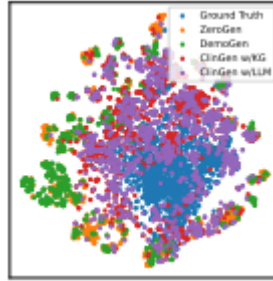


Figure 3.3: t-SNE plot example. Source: Xu et al. [158].

Central Moment Discrepancy

Central Moment Discrepancy (CMD) [164] measures how different two sets of data are by comparing their statistical moments. First, the mean of each set is calculated, and the data is centered by subtracting this mean from each data point. Then, for each moment order, such as variance and skewness, the average of the powered, centered data is calculated. The difference between the moments of the two sets is measured using the Euclidean norm:

$$\text{CMD}_k(X, Y) = \left\| \frac{1}{n} \sum_{i=1}^n (\tilde{X}_i)^{\odot k} - \frac{1}{m} \sum_{j=1}^m (\tilde{Y}_j)^{\odot k} \right\|_2 \quad (3.2)$$

Finally, the differences are added up across several moment orders to give an overall score:

$$\text{CMD}(X, Y) = \sum_{k=1}^K \text{CMD}_k(X, Y) \quad (3.3)$$

This metric is used for LLM-generated synthetic data. Adapting LLMs to a specific domain can cause a distribution shift in their generated data, since they are trained in large amounts of general-domain text [158]. CMD is a quantification of the distribution shift: a high CMD between synthetic and real data indicates a large gap between the two distributions [158].

Cosine similarity

Cosine similarity is a measure of the similarity between two vectors A and B , calculated by their normalized dot product:

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\| \|B\|} \quad (3.4)$$

In this case, the vectors are the sentence-level embeddings of the synthetic and real samples. For datasets, this metric is utilized by calculating the cosine similarity of all pairs of samples and averaging it. Low average cosine similarity of a synthetic dataset with itself is indicative of it being diverse. Low average cosine similarity of a synthetic and a real dataset indicates they are not similar, pointing towards privacy preservation by the synthetic data [58, 158].

Hausdorff distance

Hausdorff distance measures how far two subsets of a metric space are from each other as it is the greatest of all the distances from a point in one set to the closest point in the other set. In this case, using an Euclidean space (thus, Euclidean distance d), if A and B are the sentence-level embeddings of synthetic and real datasets, their Hausdorff distance is given by:

$$\text{Hausdorff Distance}(A, B) = \max \left(\max_{a \in A} \min_{b \in B} d(a, b), \max_{b \in B} \min_{a \in A} d(a, b) \right) \quad (3.5)$$

Two sets are close in Hausdorff distance if every point of either set is close to some point in the other set. Thus, the higher this distance value is, the better the privacy is preserved in a synthetic dataset, as a high value indicates that it is far from the real dataset [58].

Textual statistics

Simple counting statistics can also prove useful, such as comparing synthetic and real datasets in terms of entity quantities and frequencies [158], label quantities and frequencies [89], and unique n-grams quantities [29].

3.2.4 Datasets of synthetic unstructured clinical textual data

In Table 3.4, we present a summary of the datasets of synthetic unstructured clinical textual data found in literature. At the time of writing, we are not aware of any datasets of this nature in Portuguese.

3.2.5 Challenges

Generative LLMs have some issues that require special attention for the synthetic data to be adequate for downstream tasks. These challenges are even more important in the clinical domain, as errors and mistakes may lead to more serious problems than most other fields.

Hallucination is an issue whereby LLMs can generate information that is well-structured and grammatically accurate, but factually inaccurate or illogical. Generating false information is especially problematic in the health domain [31].

Furthermore, models have an over-reliance on training data quality. The quality and accuracy of data generated by LLMs are highly dependent on the quality and accuracy of their training data. When generating large synthetic datasets, inaccuracies on the real samples are propagated by the model [126].

A lack of domain-specific training data also affects LLMs. While synthetic data generated by LLMs can mitigate the lack of *corpora* available for clinical NLP and IE tasks, LLMs themselves can also suffer from this lack of resources. The less domain-specific data available to train the model, the less patterns and properties it will pick up, which will, in turn, reduce the quality of the synthetic samples [126].

Lastly, privacy issues are prevalent in the generation of clinical data. If the data used to pre-train and fine-tune an LLM is not properly de-identified, the application of the model may cause unintentional disclosure of private or sensitive information during the generation of the synthetic data [52, 119].

3.3 Conclusions

In this chapter, we examined the state-of-the-art in LLM-based generation of synthetic unstructured clinical textual data. We can conclude that generative LLMs have garnered significant attention in this task and various techniques have been explored to generate and evaluate this specific type of data. Synthetic data is turning into a viable alternative for real data in training NN models and other downstream tasks, as the cost of the task has been greatly reduced, while the data maintains quality and usability. However, most studies are about English data, and other languages, including Portuguese, have received little to no attention. This literature review demonstrates there is ample opportunity for this gap to be narrowed through replication of the described techniques.

Source	Language	Model	Fine-tuning	Prompt engineering	Evaluation	No. documents	Annotated	Publicly available
Peng et al. [109]	English	GatorTronGPT	290 million clinical notes + 195 English words	Iterative text generation using top-p sampling	TSTR + Turing Test	—	No	No
Tang et al. [136]	English	ChatGPT (GPT-3)	—	Prompts suggested by model, iteratively refined	TSTR + t-SNE	—	Yes (NER + RE)	No
Lorge et al. [89]	English	ChatGPT (GPT-3.5)	—	Empiric task-description prompting + in-text label after entity	TSTS + TDF-IF (labels)	3,000	Yes (NER)	Yes
Xu et al. [158]	English	ChatGPT (GPT-3.5)	—	Knowledge-infused data generation	TSTR + CMD + Entity Count + t-SNE + Cosine Similarity	5,000	No	Yes
Falis et al. [30]	English	GPT-3.5	—	Empiric task-description prompting	TSTR + Clinical Experts	9,606	Yes (EL)	Yes
Frei and Kramer [35]	German	GPT-NeoX [12]	—	Few-shot prompting, no task description	TSTS + Token Count	—	Yes (NER)	Yes
Amin-Nejad et al. [9]	English	GPT-2 small	55,404 EHRs	Structured prompts with multimodal patient context	TSTR + Perplexity	—	No	No
Kweon et al. [77]	English	GPT-3.5 turbo	—	Clinician-reviewed prompts to convert case reports into synthetic clinical notes	Perplexity + Clinical Experts	158,000	No	Yes

Table 3.4: Datasets of synthetic unstructured clinical textual data.

Chapter 4

Methodology

In this chapter, we begin by defining our research hypothesis. Then, we summarize our approach to generating and annotating a dataset of synthetic clinical notes in European Portuguese, which we named LusoClin. Afterward, we describe the generation of LusoClin, which includes: justifying the selection of the employed LLM and the exclusion of various other options; presenting the datasets chosen for fine-tuning and to be used as examples during prompting; outlining the pre-processing of these datasets; describing how the selected LLM was fine-tuned; detailing the prompt engineering process, which included a prompt evaluation done by medical students; and explaining the generation and post-processing of the data. Lastly, we describe the NER annotation task.

4.1 Approach

We begin by defining our research hypothesis as follows:

RH: An open-source LLM can generate a large-scale European Portuguese dataset of synthetic clinical notes with enough utility, fidelity, diversity, and privacy to replace real data in downstream applications.

Our approach was designed based on reviewed literature and consists of using an open-source generative LLM to generate LusoClin, an European Portuguese dataset of synthetic clinical notes, annotating it at the NER level with a BERT-based model, and evaluating it using human and automatic methods.

Figure 4.1 delineates the general architecture of the approach we used to test our research hypothesis.

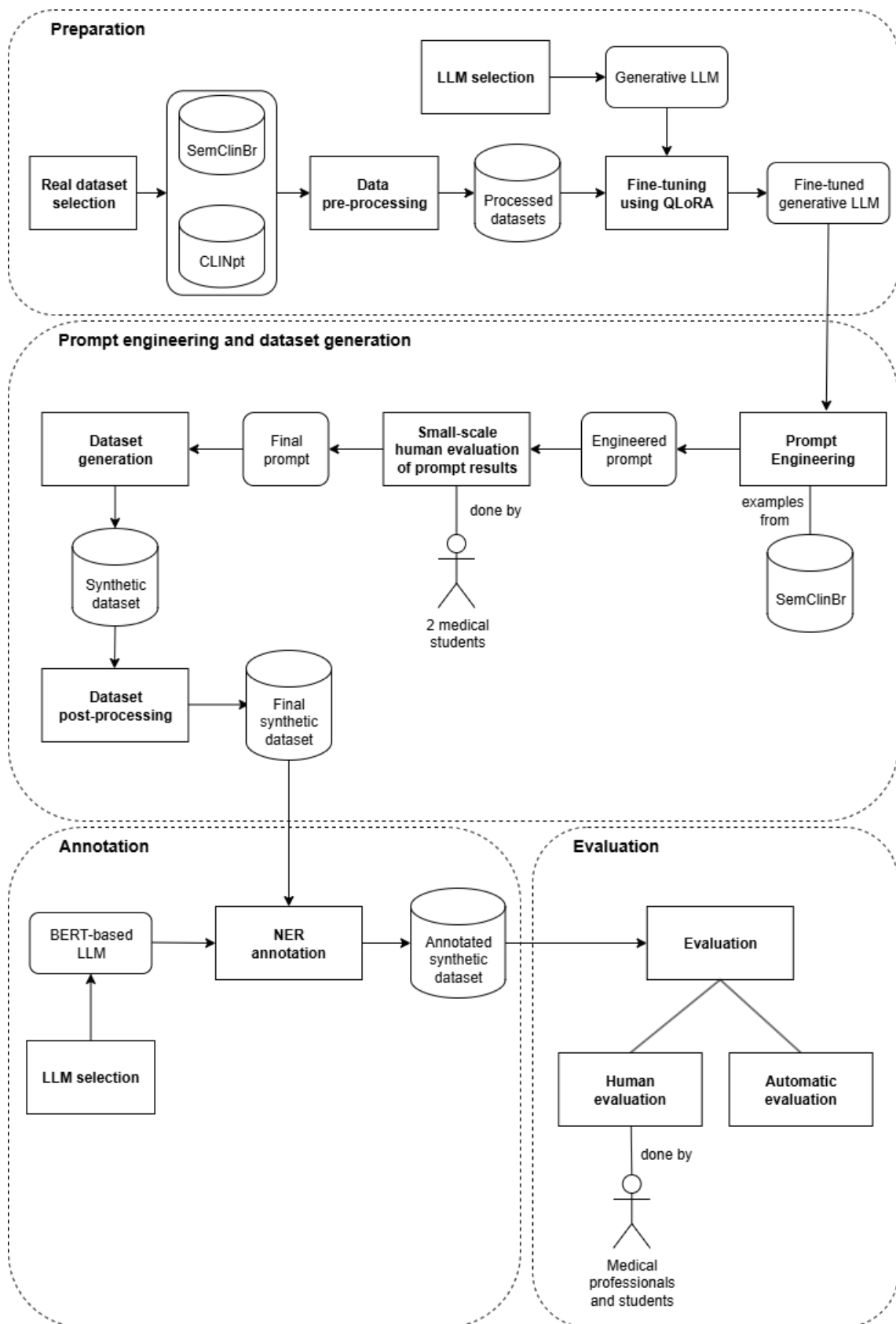


Figure 4.1: Architecture of the approach used to generate, annotate, and evaluate LusoClin.

4.2 Preparation

In this section, we detail our process to prepare the generation of LusoClin. We describe the selection of the generative LLM, the selection of real datasets for fine-tuning and prompt engineering, the pre-processing of these datasets, and the fine-tuning of the selected LLM.

4.2.1 LLM selection

To generate LusoClin, we needed to select an LLM. We opted for open-source due to four reasons: firstly, we avoided the risk of passing sensitive information to models run by third-parties; secondly, we could run the model locally, allowing us greater control over data passed to it; thirdly, with publicly available model weights, we could fine-tune the model to help fulfill our specific task; lastly, since these models can be freely utilized by researchers, it is important to investigate their capabilities. Several other requirements were weighed when choosing the model: it needed to be a generative LLM; its context window had to be larger than the biggest individual document used for fine-tuning and larger than prompts containing several documents as examples, so 2,048 tokens would be a safe threshold; it had to fit into our GPU and storage, which excluded models with too many parameters; it had to be able to generate text in Portuguese, specifically European Portuguese, which meant its training data had to include Portuguese text; it needed to have instruction-tuning, so it could be used in a dialog (prompt/response) format.

Given these criteria, we opted for Meta’s Llama 3 [140], specifically Llama 3.2 with 3 billion parameters and instruction tuning, available through the HuggingFace API, under the name ‘Llama-3.2-3B-Instruct’¹.

Llama 3.2 is a large language model developed by Meta, as part of the Llama 3 herd of models. Figure 4.2 depicts their architecture. Llama 3.2 is an auto-regressive language model that uses an optimized Transformer architecture. The tuned versions use supervised fine-tuning and RLHF [140]. The architecture is very similar to Llama and Llama 2, with performance gains coming from improvements in data quality and diversity, as well as from an increased training scale [140]. The model supports several languages, including Portuguese. The multilingual training data comes, primarily, from human annotations and datasets from other NLP tasks. This training is performed by branching off the pre-training run and continuing to pre-train on a data mix that consists of 90% multilingual tokens [140]. We chose the highest-parameter instruction-tuned version available that our GPU could handle.

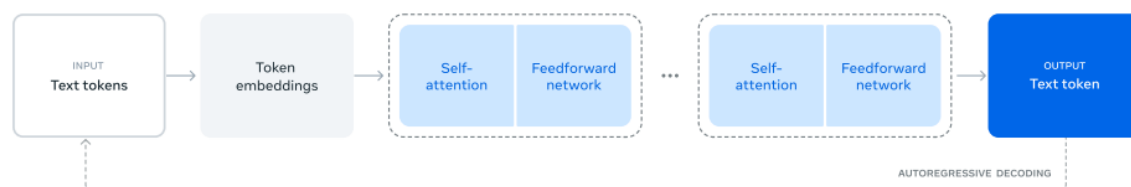


Figure 4.2: Llama 3 architecture. Source: Dubey et al. [140].

¹<https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct>

We considered other models but excluded them for a variety of reasons, as Table 4.1 demonstrates.

LLM	Description	Reason(s) for exclusion
Mistral family [66]	Open-source, Transformer-based, uses grouped-query attention for faster inference, coupled with sliding window attention	Unable to properly generate Portuguese text
Gemma family [139]	Open-source, Transformer-based, uses multi-head attention and rotary position embeddings	Instruction-tuned models only available in English
Bode [41]	Llama 2 fine-tuned on Brazilian Portuguese instructions dataset	Brazilian Portuguese
DrBode [107]	ChatBode [38] fine-tuned on Brazilian Portuguese instructions dataset, with 50% medical data	Brazilian Portuguese; medical fine-tuning performed with LLM-translated data
Sabiá [111]	Llama 1 fine-tuned on Brazilian Portuguese dataset	Brazilian Portuguese
Cabrita [78]	Llama-based model additionally pre-trained on Brazilian Portuguese dataset	Brazilian Portuguese
Glória [87]	GPT-Neo [13] (GPT-3 replica) architecture pre-trained on European Portuguese <i>corpus</i>	No instruction-tuned version available

Table 4.1: LLMs and reasons for exclusion.

4.2.2 Real dataset selection

Since the goal of this project was to generate synthetic clinical data in Portuguese, we leveraged some of the existing high-quality Portuguese clinical datasets for various phases of our process. We elected to use SemClinBr [104] and CLINpt [86]. Both are publicly available; however, to access SemClinBr, one must first fill out the request form available in their GitHub repository².

SemClinBr [104] is a dataset containing 1,000 Brazilian Portuguese clinical notes of four different narrative types: admission notes, ambulatory (or medical appointment) notes, cardiology discharge summaries, and nursing notes. Admission notes explain the cause for a patient’s admission and outline their situation; ambulatory notes are written by physicians to detail what happened during an appointment with a patient; discharge summaries are written upon a patient’s discharge from the institution and summarize their stay; nursing notes are written by nurses and report a patient’s situation, symptoms, tests, and medication, over a span of time.

The data comes from two different sources: 2,094,929 documents from a group of hospitals in Brazil generated between 2013 and 2018; 5,617 entries from a university hospital in Brazil

²<https://github.com/HAILab-PUCPR/SemClinBr>

generated in the period between 2002 and 2007. Of these, 1,000 documents were selected and professionally de-identified. Table 4.2 lists the medical specialties present in the dataset and their respective number and percentage of notes. Afterward, they were annotated at the NER level, with each annotation being tagged with one of 87 different semantic types (such as ‘Body Location or Region’, ‘Disease or Syndrome’, or ‘Therapeutic or Preventive Procedure’), plus ‘Abbreviation’ and ‘Negation’. They were also annotated with two relation types between entities: ‘associated_with’, and ‘negation_of’.

Specialty	Number	%
Cardiology	260	23.48
Nephrology	157	14.18
Orthopedics	126	11.38
Not defined	122	11.02
Surgery (general)	61	5.51
Neurology	45	4.06
Neurosurgery	32	2.89
Dermatology	23	2.08
Ophthalmology	22	1.99
Endocrinology	19	1.72
Gastroenterology	16	1.45
Otolaryngology	14	1.26
Pneumology	11	0.99
Others	92	8.31

Table 4.2: Number and percentage of notes for each medical specialty in the SemClinBr dataset. Medical specialties with fewer than ten occurrences were grouped into the “Others” category. Source: Oliveira et al. [104].

The article [104] lists the number of notes for each type: 148 admission notes, 506 ambulatory notes, 126 discharge summaries, and 232 nursing notes. However, we are not given the type of each individual note, and these values do not add up to 1,000. Therefore, we followed the authors’ suggestion and used ChatGPT (GPT-4) to classify them, using the following prompt: ‘Classifica a seguinte nota clínica com um destes tipos: nota de internamento, registo de consulta, nota de alta, nota de enfermagem.’, which can be translated to English as ‘Classify the following clinical note using one of these types: admission note, ambulatory note, discharge summary, nursing note.’. The English version was not used and is presented merely due to the language of this document. We obtained these results: 52 admission notes, 544 ambulatory notes, 132 discharge summaries, and 272 nursing notes, which we used for the rest of this work. Figure 4.3 shows the distribution of word counts per type of note, and Figure 4.4 shows the distribution of sentence counts per type of note.

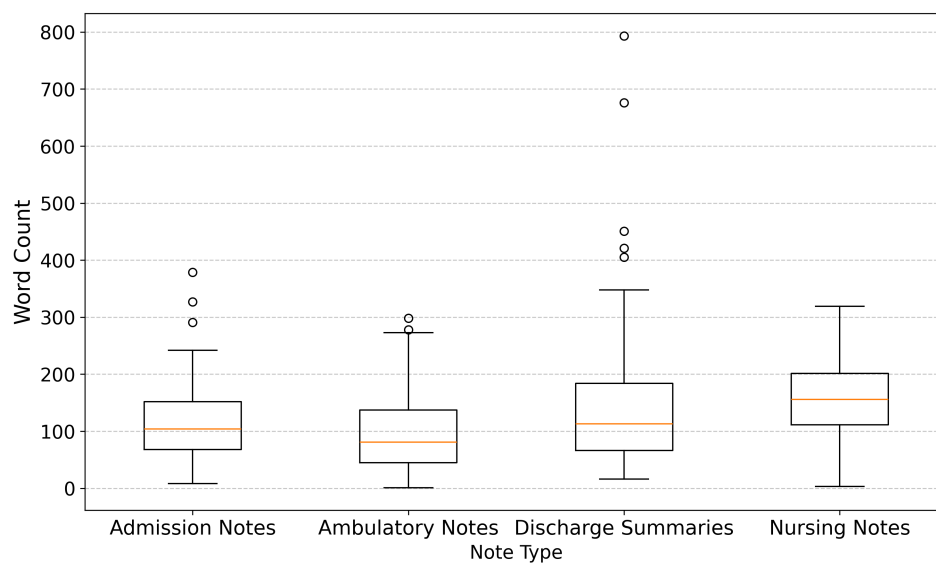


Figure 4.3: Different types of clinical notes have different word distributions.

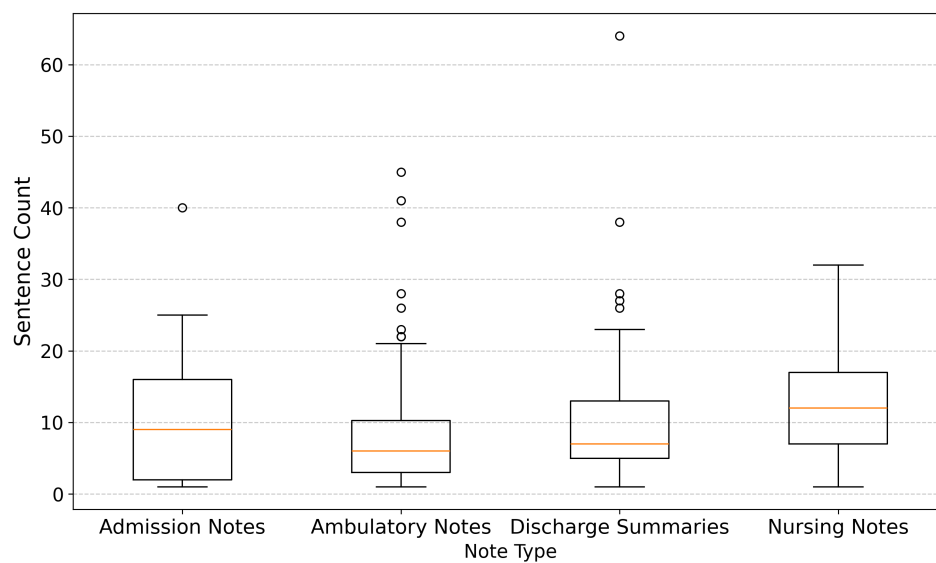


Figure 4.4: Different types of clinical notes have different sentence distributions.

Each note is represented by an *XML* file with three sections: ‘text’, ‘annotations’, and ‘relations’, as depicted by the example in [Listing 4.1](#):

```

1 <?xml version='1.0' encoding='UTF-8'?>
2 <ANNOTATIONS>
3 <TEXT>pós-amputação traumática de 3º FD da mão esquerda.
4 rx sem alterações.
5 fo cicatrizada.
6 adm pouco limitado.

```

```

7 cd:
8 retorno ambulatorial.
9 oriento fst e ganho de adm.
10 r3 Vital Brasil</TEXT>
11 <TAGS>
12 <annotation id="1379" tag="Diagnostic Procedure" start="52" end="54" text="rx" abbr
   ="" />
13 <annotation id="1380" tag="Abbreviation" start="110" end="112" text="cd" abbr="" />
14 <annotation id="1381" tag="Health Care Activity" start="115" end="122" text="
   retorno" abbr="" />
15 <annotation id="1382" tag="Health Care Related Organization" start="123" end="135"
   text="ambulatorial" abbr="" />
16 <annotation id="1383" tag="Health Care Activity" start="138" end="145" text="
   oriento" abbr="" />
17 <annotation id="1384" tag="Therapeutic or Preventive Procedure|Abbreviation" start=
   "146" end="149" text="fst" abbr="fisioterapia" />
18 <annotation id="1385" tag="Abbreviation|Finding" start="72" end="86" text="fo
   cicatrizada" />
19 <annotation id="1386" tag="Abbreviation|Finding" start="89" end="107" text="adm
   pouco limitado" />
20 <annotation id="1387" tag="Abbreviation" start="152" end="164" text="ganho de adm"
   />
21 </TAGS>
22 <RELATIONS>
23 <rel annotation1="1384" annotation2="1383" reltype="associated_with" />
24 </RELATIONS>
25 </ANNOTATIONS>

```

Listing 4.1: SemClinBr example. Source: Oliveira et al. [104].

Lopes et al. [86] created a dataset containing 281 European Portuguese neurology clinical cases, with manually annotated named entities. As done by Schneider et al. [120], we refer to it as CLINpt throughout this dissertation.

The data comes from numbers 1 and 2 of Volume 17 of the clinical journal Sinapse [7], published by the Portuguese Society of Neurology. The notes were annotated at the token level, with each token being PoS-tagged, lemmatized, and IOB2-tagged. The IOB (inside, outside, beginning) format [114] is a token tagging format. The IOB2 variation tags the first token of an entity using ‘B-[label]’ and all following tokens of the same entity with ‘I-[label]’. Any token that isn’t part of an entity is tagged with ‘O’. In CLINpt, the IOB2 tags include labels such as ‘Characterization’, ‘Evolution’, and ‘Therapeutics’. Table 4.3 provides an example of an annotated excerpt from a case.

Token	POS Tag	Lemma	IOB Tag
de	prp	de	O
66	num	66	O
anos	n	ano	O
,	punc	,	O
com	prp	com	O
antecedentes	n	antecedente	B-DT
de	prp	de	O
dislipidemia	n	dislipidemia	B-C
e	conj-c	e	O
síndrome	n	síndrome	B-C
depressiva	adj	depressivo	I-C
,	punc	,	O
começou	v-fin	começar	O
por	prp	por	O

Table 4.3: Annotated excerpt from CLINpt. Source: Lopes et al. [86].

Figure 4.5 and Figure 4.6 show CLINpt’s word and sentence count distributions, respectively.

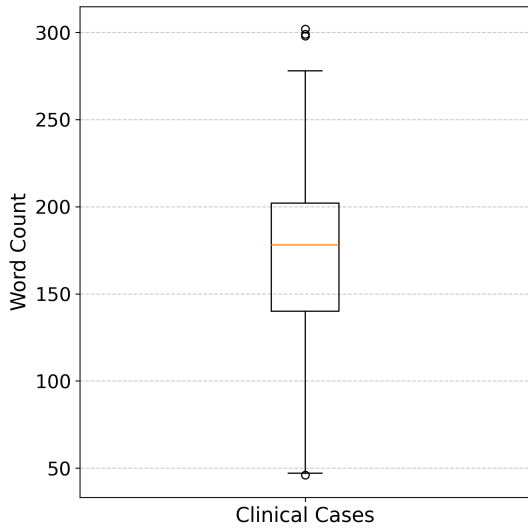


Figure 4.5: Word count distribution for CLINpt’s clinical cases.

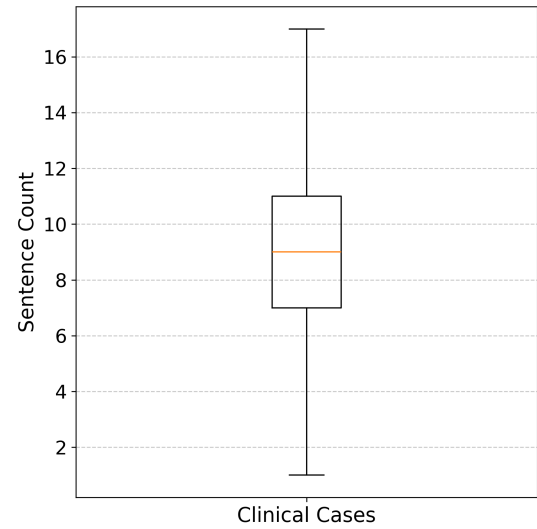


Figure 4.6: Sentence count distribution for CLINpt’s clinical cases.

Each note is represented by a *TXT* file containing the unstructured text and an *XLSX* file with the structure shown in Table 4.3.

SemClinBr was used to fine-tune the LLM, to provide examples during prompting, and to help during evaluation. CLINpt was only used to fine-tune the model, as it is a single-domain dataset (neurology).

4.2.3 Data pre-processing

As explained in [Section 2.4.1](#), transfer learning allows LLMs to be easily fine-tuned to improve on a specific NLP task or knowledge domain, while keeping the knowledge obtained during pre-training. Open-source LLMs have publicly available weights, meaning any user has the possibility to fine-tune them. Therefore, we fine-tuned our selected generative LLM to attempt to help improve its clinical NLP abilities, using the data from SemClinBr and CLINpt.

Following studies [9, 109], we made some changes to all 1,000 SemClinBr documents and all 281 CLINpt documents. For SemClinBr, we extracted the ‘text’ section of the *XML* documents, which contains the clean unstructured text of each clinical note. For CLINpt, this was not necessary, as the text is already available in *TXT* files. For each note of both datasets, we performed three operations, depicted in [Table 4.4](#). We did not remove low count tokens, as there were few documents and a low count did not necessarily mean a token was irrelevant to clinical patterns. Since none of the documents were anywhere near the 128k token context window of our selected LLM, we did not need to exclude any, on either dataset.

Operation	Input Example	Output Example
Lowercase all characters	‘CURATIVO COM ASPECTO LIMPO E SECO’	‘curativo com aspecto limpo e seco’
Replace characters with diacritics with their non-diacritic counterparts	‘sob consulta cardiológica para avaliação’	‘sob consulta cardiologica para avaliacao’
Remove delimiters and other punctuation	‘abertura ocular normal, agitado.’	‘abertura ocular normal agitado’
All operations combined	‘LÚCIDO, ABERTURA OCULAR ESPONTÂNEA.’	‘lucido abertura ocular espontanea’

Table 4.4: Data pre-processing operations.

4.2.4 Fine-tuning using QLoRA

In order to fine-tune the model we selected, we applied a technique called Quantized Low-Rank Adaptation (QLoRA) [26], which significantly reduces the necessary resources while achieving great results.

Low-Rank Adaptation (LoRA) [60] is a fine-tuning method designed to be more efficient than traditional fine-tuning. Traditional fine-tuning involves updating all the parameters of a model, which can be computationally expensive and storage-intensive, especially for massive models with billions of parameters, such as LLMs. LoRA freezes the original weight matrices and introduces lightweight low-rank adaptations of them in their place. The adaptations can be much more quickly and resource-efficiently fine-tuned. During inference, the original model weights and the fine-tuned low-rank adaptations are combined, allowing the model to perform well on the new task without compromising its original pre-trained capabilities. If x is an input value, h is the output

vector, W_0 is the original pre-trained weight matrix, and ΔW is the update rule, regular fine-tuning would be represented by:

$$h = W_0x + \Delta Wx \quad (4.1)$$

With LoRA, the update is constrained by being turned into BA , where A and B are two low-rank adaptations of W_0 , with 0.1% of the size, or less. W_0 is frozen and A and B contain trainable parameters. Thus, fine-tuning becomes:

$$h = W_0x + BAx \quad (4.2)$$

Figure 4.7 illustrates this adaptation. LoRA allows faster training and with significantly fewer GPUs [60].

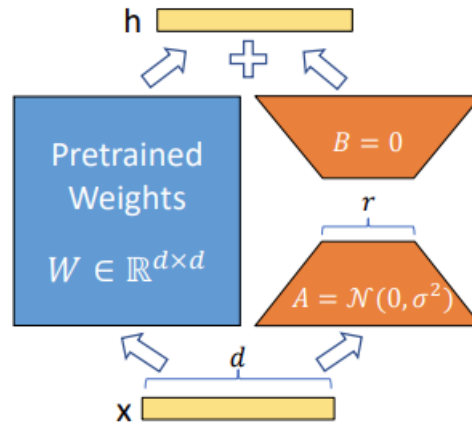


Figure 4.7: LoRA method. Source: Hu et al. [60].

QLoRA builds on LoRA by adding another layer of efficiency, through the use of the 4-bit NormalFloat (NF4) floating-point representation and Double Quantization (DQ). NF4 is a floating-point representation format that allows storage of floating-point values using only 4 bits. In the context of ML, quantization is the process of constraining the number of bits that represent a model's parameters. DQ involves two quantizations: first, the model's parameters are quantized to the NF4 representation format. The metadata for this quantization is stored so the parameters can be restored for use during computation (necessary for the model to maintain precision). Then, the metadata is also quantized, in this case to an 8-bit format. This method greatly reduces memory requirements, allowing models with billions of parameters to be fine-tuned on a single GPU [26].

To perform the fine-tuning, we used torchtune³, a PyTorch [108] library for easily authoring, fine-tuning and experimenting with LLMs, due to its support of Llama models, QLoRA, and custom datasets. The torchtune library uses configuration files in *YAML* format to set up fine-tuning, which allows for great control over training specifications.

³<https://github.com/pytorch/torch tune>

The model was fine-tuned using the AdamW optimizer [90], with a learning rate of $3e^{-3}$ and a weight decay of 0.01. Training was carried out over 4 epochs with a batch size of 4 and gradient accumulation over 4 steps. Additionally, 100 warmup steps were employed. All pre-processed documents from SemClinBr and CLINpt were used. The fine-tuning was performed on a Virtual Machine that included 100GB of RAM, 100GB of storage, and a single Nvidia RTX 3090 graphics processing unit with 24GB of vRAM.

4.3 Prompt engineering and dataset generation

In this section, we describe the prompt engineering phase, a small-scale human evaluation of prompt results, and the generation and post-processing of LusoClin.

4.3.1 Prompt engineering

Due to the exploratory nature of this task, we began by tuning the prompt through trial and error, at the level of characteristics identifiable to the layman’s eye, until we reached responses that consistently resembled clinical notes. For this, we opted for an approach of combining methods seen in the literature and explained in [Section 3.2.2](#), and tuning the model’s temperature.

At first, we made simple requests, prompting the model to act like a Portuguese doctor and generate a clinical note that would describe a patient’s crossing through a hospital. As responses were provided, we adjusted the prompt by describing the task in more detail, making specific requests to avoid common errors the model was making, and adding examples from SemClinBr (few-shot prompting).

Then, we started recording the prompts from this engineering process and the model’s responses. These records attest to our focus on addressing specific issues found in the model’s answers. These included several specific requests, such as maximum or exact word, sentence, and paragraph counts, running text with no subdivisions, no sections such as ‘Age: 52 years old’, no signatures, no listing of examples, no repetition of consecutive sentences, no constant repetition of the word ‘patient’, and no augmentation of the provided examples. These specifications were variously combined in an attempt to reach a reasonable balance between error-fixing and not overwhelming the model.

We refrained from explaining the details of each note type, allowing the model to autonomously learn from the examples. Along the way, we also attempted to switch the prompt language to English, to no success. Furthermore, we settled on following the approach of Xu et al. [158] and injected each prompt with examples of notes that contained a specific ‘Disease or Syndrome’ entity that was randomly chosen from the ‘Disease or Syndrome’ entities in SemClinBr. We picked this entity type due to its potential to provide diversity in the content of the examples.

This exploratory process helped us grasp the general capabilities of the LLM for clinical synthetic data generation. We found several limitations with the model, which defined this prompt engineering phase and shaped the prompt that resulted from it. This analysis is also valuable to help future research determine whether this model can perform their desired clinical tasks, and

what shortcomings need to be addressed to improve its behaviour. We came across the limitations described in [Table 4.5](#).

Prompt challenges	Example of prompt response
Giving too many specific instructions in a prompt seemed to confuse the model, forcing us to leave out some necessary specifications, and making it challenging to address specific problems with the generated notes.	Cannot be shown using an example. When requesting too many specificities in one prompt such as note length, medical info, error correction, among others, the response would ignore many of them.
Despite rejecting some other LLMs due to them being specifically Brazilian Portuguese, the model also couldn't avoid using some Brazilian Portuguese terms, even when specifically prompted to use European Portuguese.	'[...] A nota médica foi atualizada e o paciente foi <i>liberado</i> para continuar o tratamento em casa. O paciente foi acompanhado por um enfermeiro durante a <i>internação</i> e foi mantido em <i>contato</i> com o médico para <i>monitorar</i> a recuperação. O paciente foi <i>liberado</i> para voltar a <i>suas</i> atividades normais em 10 dias. [...]
The model ignored most requests to limit the size of generated clinical notes (which forced us to use the maximum generated tokens configuration).	Not adequate to be displayed here, due to size.
The model ignored most specific requests for the generated notes to have complete information.	For example, many prompts did not include vital signs, even though they are standard, especially in admission and nursing notes.
Examples in European Portuguese helped reduce Brazilian Portuguese terms in the generated notes, when combined with low temperature. However, the only publicly available real clinical European Portuguese dataset, CLINpt, only contains clinical cases. If used for examples, the model steered away from generating clinical notes, generating cases instead.	'Ocorrência hospitalar de um homem de 65 anos que apresenta dor abdominal severa e diúrese, após ter sido atingido por uma arma de fogo. Ao ser recebido no hospital, apresenta pressão arterial de 180/100 mmHg e frequência cardíaca de 120 batimentos por minuto. [...]
The model repeated sentence structures and very often started every sentence with 'O paciente' ('The patient'). This gave many generated notes a clear synthetic feel, even if the generated information was correct and useful.	'[...] O paciente refere dor em dorso nasal [...] O paciente refere também dor de cabeça, que ocorre todos os dias, e que tem início a cada 2 horas, na manhã e à noite. O paciente refere também [...]
Sometimes, the model generated a prompt or a continuation of the given prompt, instead of fulfilling the request.	'[...] a nota de enfermagem não deve ser apenas uma repetição de informações que já foram discutidas anteriormente. Em vez disso, ela deve ser [...]
Prompts in English that included a request for the response to be in European Portuguese were very ineffective. This was tried to see if the model better heeded the array of instructions if they were in English.	'56 years old male patient, arrived at the hospital 16:30h. Trauma by vehicle, multiple injuries trauma iliac bone fracture, damage to the aorta and splenic artery, damage to the kidneys, lung contusion [...]

Table 4.5: Conclusions about prompt challenges the model faced.

The recorded prompts and responses produced during this stage, along with the temperature for each prompt, are available in the provided repository [37].

Table 4.6 shows the engineered prompt template that resulted from this phase and its English translation. The English version was not used and is presented merely due to the language of this document.

Original Prompt (Portuguese)	English Translation
Escreve, em português europeu, [Clinical Note Type] com 10 frases, 130 palavras, e com apenas 1 parágrafo. [Clinical Note Type] tem que estar em texto corrido, sem qualquer subdivisão. Não incluas nenhuma secção do estilo 'Idade: 52 anos'. Não repitas a palavra 'paciente' várias vezes. Não repitas frases. A tua resposta deve incluir apenas [Clinical Note Type].	Write, in European Portuguese, [Clinical Note Type] with 10 sentences, 130 words, and only 1 paragraph. [Clinical Note Type] must be in running text, without any subdivision. Do not include any 'Age: 52'-style sections. Don't repeat the word 'patient' several times. Don't repeat sentences. Your answer must include only [Clinical Note Type].
Alguns exemplos são: [Examples with one randomly chosen 'Disease or Syndrome' entity]	Some examples are: [Examples with one randomly chosen 'Disease or Syndrome' entity]
Não dês mais exemplos. Não desenvovas nenhum destes exemplos.	Don't give any more examples. Don't elaborate on any of these examples.

Table 4.6: Prompt template after *prompt engineering* and an English translation.

Where [Clinical Note Type] was one of: 'nota de internamento', 'registo de consulta', 'nota de alta', or 'nota de enfermagem', which mean 'admission note', 'ambulatory note', 'discharge summary', and 'nursing note', respectively, following SemClinBr's clinical note types. As the types in the SemClinBr article [104] are in English, two medical students helped correctly translate them to European Portuguese. On each prompt, the number of provided examples was equal to the number of available clinical notes of the type being generated that contain the randomly picked 'Disease or Syndrome' entity, up to a maximum of four examples. So, for instance, if we were prompting for the generation of an ambulatory note, the randomly picked entity was 'flu', and there are five ambulatory notes in SemClinBr that mention 'flu', four of those would be randomly chosen as examples. If there is only one ambulatory note that mentions 'flu' in SemClinBr, that note would be the only example. The goal of prompt requests that seem too specific was to address particular errors commonly made by the model. Requests to limit the note's word and sentence counts were made to counter the model's tendency to generate overly large notes. Values of 130 words and 10 sentences were chosen based on the average note size of SemClinBr. After empirically testing multiple temperature values throughout this process, we settled on a temperature of 0.6.

Given the prevalence of some errors, we performed a rule-based adjustment on each response to obtain the clinical notes, with the following rules: removing sentences with the words ‘frase’ (sentence), ‘frases’ (sentences), ‘parágrafo’ (paragraph), ‘parágrafos’ (paragraphs), ‘texto’ (text), as the model would sometimes attempt to replicate the prompt and generate instructions, before beginning the note; removing repeated sentences and keeping only their first instance, as the model would sometimes finish notes with an endless repetition of the last sentence, while the rest of the note was valid; removing the last sentence if it was incomplete, as the model would frequently generate text endlessly, only being limited by the ‘maximum new generated tokens’ configuration, causing the last sentence to be abruptly interrupted; removing isolated numbered points such as ‘1.’, as the prompt specifications would sometimes cause the model to number the generated sentences.

4.3.2 Small-scale evaluation of prompt results

In order to determine whether our prompt required further engineering, we performed a small-scale evaluation with the assistance of two final-year medical students, as shown in [Figure 4.8](#). We randomly selected 16 synthetic notes (4 of each type) and 8 SemClinBr notes (2 of each type). Each of the students read and analysed a different half of the set, so, 8 synthetic notes, 2 of each type, and 4 SemClinBr notes, 1 of each type. They were not informed of the type of each note, nor if they were real or synthetic. They answered the following questions:

- Q1. What is the note’s type (admission note, ambulatory note, discharge summary, nursing note)?
- Q2. Is the note real or synthetically generated? Why?
- Q3. Using a 5-point Likert scale, where 1 means ‘Very Little’ and 5 means ‘Very Much’, rank the confidence of your answer to the previous question.
- Q4. Using a 5-point Likert scale for each parameter, where 1 means ‘Very Poor’ and 5 means ‘Very Good’, evaluate the note on the following parameters: medical information, structure, terminology, completeness. *Medical information* indicates the accuracy and coherence of the medical circumstances described by a note. *Structure* indicates if a note follows its type’s usual order of information. *Terminology* indicates if the terms used throughout the note are the ones commonly used by physicians and nurses to describe similar situations. *Completeness* indicates whether the note contains all the information its specific type entails.

Question 1 served a double purpose: evaluating our model’s ability to generate clinical notes that resemble their type, and assessing ChatGPT’s ability to classify a clinical note’s type. We recall here that we used ChatGPT to classify the notes from SemClinBr, following a suggestion by the authors of SemClinBr. Question 2 was a Turing Test to gauge the synthetic data’s similarity to real data, and allowed the medical students to openly provide their judgement on positive and negative stand-out features of a note. Question 3 helped better ascertain the level of realism of

the synthetic notes. Question 4 helped specify what was sub-par and what was good about the synthetic notes.

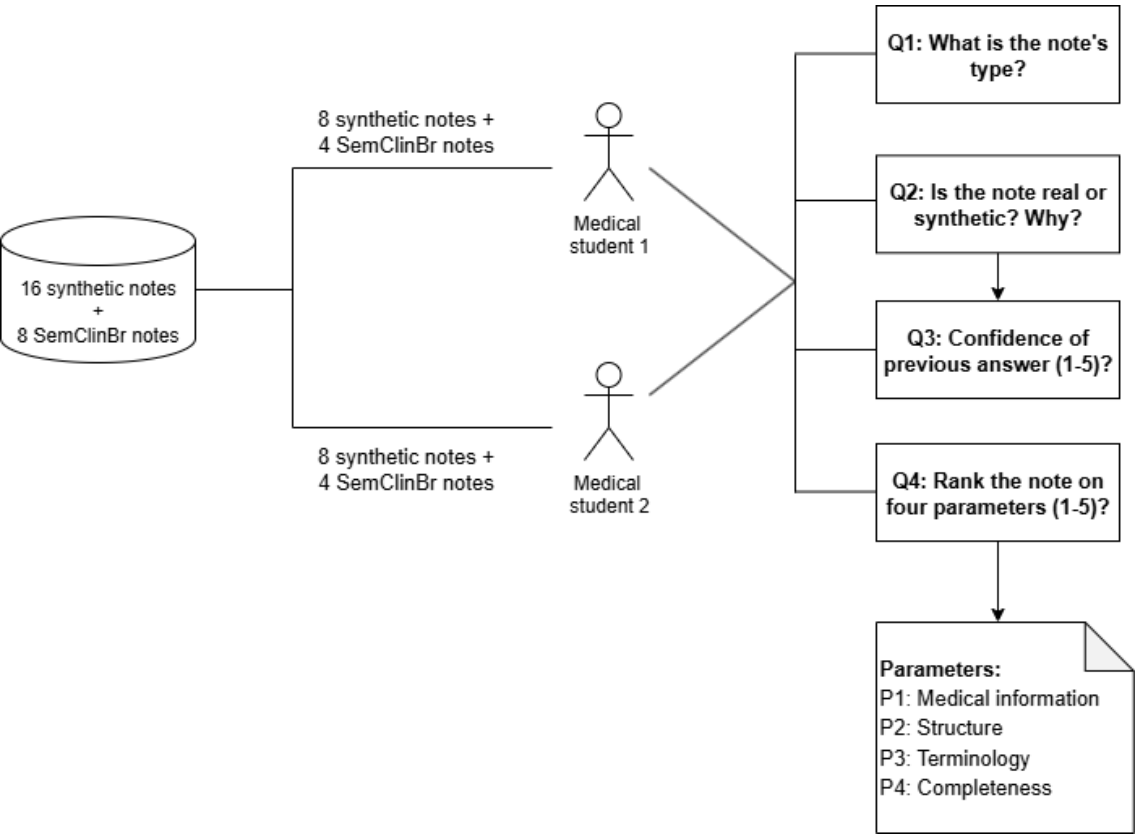


Figure 4.8: Diagram of small-scale prompt evaluation.

Despite the reduced scale of this evaluation in comparison with LusoClin’s intended size, it allowed us to evaluate the need of changes to the engineered prompt before proceeding to the dataset generation. The results are presented and analysed below. Table 4.7 displays the type guessing results, showing how often the students agreed with ChatGPT’s classification of real (SemClinBr) notes and with our model’s classification of its generated notes, in terms of note type. Table 4.8 shows the results regarding the Turing tests (questions 1, 2, and 3). ‘Assessed synthetic’ and ‘Assessed real’ show the times the students evaluated a note as synthetic and real, respectively. The percentage next to it indicates how often their evaluations were accurate. Each has a corresponding ‘Avg. confidence’ column, displaying the average confidence of the corresponding evaluations. Every column is divided by real and synthetic notes. Table 4.9 displays the average scores of the four evaluation parameters (question 4). ‘Note type avg.’ is the average of all parameters for that note type. Again, every column is divided by real and synthetic notes.

Note type	Source	N	Agreed with note type	%
Admission note	Real	2	0	0%
	Synthetic	4	4	100%
Ambulatory note	Real	2	1	50%
	Synthetic	4	3	75%
Discharge summary	Real	2	2	100%
	Synthetic	4	3	75%
Nursing note	Real	2	2	100%
	Synthetic	4	2	50%
Total	Real	8	5	63%
	Synthetic	16	12	75%

Table 4.7: Type guessing results.

Investigating the type guessing on synthetic notes, we see that the medical students agreed with the model’s intentions 75% of the time. Only nursing notes obtained a lower result of 50%. On the other hand, they agreed with ChatGPT on the real notes only 63% of the time. This could mean LLMs are better at successfully imprinting synthetic notes with distinguishable patterns or terms that make a clinical narrative type identifiable by medical personnel, then they are at classifying existing notes. It could also mean that ChatGPT is not a reliable clinical narrative type identifier for the Portuguese language. However, the sample size is too small for any definitive conclusions.

Note type	Assessed synthetic		Avg. confidence		Assessed real		Avg. confidence	
	Real	Synthetic	Real	Synthetic	Real	Synthetic	Real	Synthetic
Admission note	2/2 (0%)	2/4 (50%)	3.5	3.5	0/2 (0%)	2/4 (50%)	-	3
Ambulatory note	0/2 (100%)	2/4 (50%)	-	4.5	2/2 (100%)	2/4 (50%)	4.5	4
Discharge summary	1/2 (50%)	4/4 (100%)	4	4	1/2 (50%)	0/4 (100%)	4	-
Nursing note	1/2 (50%)	3/4 (75%)	3	4.33	1/2 (50%)	1/4 (75%)	4	4
Total	4/8 (50%)	11/16 (69%)	3.5	4.08	4/8 (50%)	5/16 (69%)	4.17	3.67

Table 4.8: Turing test results.

The Turing Test constituted the most important part of this process. If synthetic notes can pass as real, it is a great pointer towards their viability as substitutes of real data in real-world scenarios, thus greatly strengthening our confidence in our prompt.

The synthetic notes only tricked the medical students 31% of time. Moreover, the average confidence with which they thought a synthetic note was real was slightly lower than the average confidence with which they correctly said a note was synthetic (3.67 and 4.08, respectively). Nonetheless, it is pivotal that we analyse the reasons why the medical students thought notes were synthetic. While in a couple of cases they describe the notes as containing incorrect medical information and non-existent terminology, most synthetic notes were correctly identified as synthetic through obvious markers of non-human generation, even if the described medical circumstances

made sense. These markers included unnatural sentence organization, abnormal or non-sensical repetition of information, inclusion of unnecessary information, and confusing structure.

It is also worth noting that 50% of real notes were classed as synthetic, with an average confidence of 3.5. Our interpretation is that, even if the synthetic notes are scarcely able to pass as real, it is rarely due to low-quality medical information or terminology, meaning they hold more value than this statistic appears to imply. This conclusion is corroborated by the parameter results, which we analyse in the remainder of this section.

Note type	Medical info		Structure		Terminology		Completeness		Note type avg.	
	Real	Synt	Real	Synt	Real	Synt	Real	Synt	Real	Synt
Admission note	4	4	3.5	3.5	4	4.25	3	3.5	3.63	3.81
Ambulatory note	5	4.5	4.5	3.25	5	4.25	2.5	3	4.25	3.75
Discharge summary	5	3.25	3.5	3	5	3.75	4	2.75	4.38	3.19
Nursing note	4.5	2.75	3	2.75	5	3.5	3.5	2.5	4	2.88
Parameter Avg	4.63	3.63	3.63	3.13	4.75	3.94	3.25	2.94	4.07	3.41

Table 4.9: Evaluation parameter results.

Looking at the synthetic notes' parameter results, we can gather that the model was better at generating admission and ambulatory notes than nursing notes and discharge summaries. They had more correct medical information, were better structured and used more realistic terminology. We may speculate on the causes of this difference. Nursing notes and discharge summaries tend to be more diverse and contain more information, as each note reports on several types of events (symptoms, tests, medication, among others). Furthermore, nursing notes are more chaotic by nature, which can make it more challenging for the model to understand the underlying patterns that make up a typical nursing note. Discharge summaries can be more easily considered faulty or lacking, as they summarize a patient's stay in an institution, which means there is more room for error among the vast and diverse information provided.

In terms of completeness, the model struggled across the board. However, the real notes only outperformed the synthetic notes on this parameter by 0.31 points, meaning this could be a general problem with clinical notes, not a problem with the model's generation abilities.

Looking solely at admission and ambulatory notes, the synthetic notes are on par with the real ones, even outperforming them on completeness. Only the structure of synthetic ambulatory notes falls short. This is a very positive result, as it means that these synthetic types appear to match the quality of real notes, or, in other words, are realistic. If a note is realistic, we can infer that it has good utility and fidelity, and that the prompt that generated it is adequate.

Shifting our attention to nursing notes and discharge summaries, real notes strongly surpass synthetic notes on medical information, terminology, and completeness, and slightly surpass them on structure.

Analysing the parameter results of the notes as a whole, we conclude that real notes outperformed our synthetic notes on every parameter. However, all synthetic parameter averages were outperformed by one point or less, and most averages were significantly lowered by synthetic nursing notes and discharge summaries. This means we would be more precise to attempt general statements on a type-by-type basis: synthetic admission and ambulatory notes seem close to the quality of real ones, while nursing notes and discharge summaries seem farther.

The first conclusion is that it would be necessary to adjust the prompt for nursing notes and discharge summaries, perhaps better detailing the specifics of these narrative types to help the model. However, this had been attempted before this evaluation and before prompts and responses were logged. As mentioned earlier in this section, the model was unable to deal with too many specific requests, which, in this case, caused responses to largely ignore details about what each narrative type generally requires. This was clear, even to our untrained eye, which is why we refrained from that approach and relied on examples. Perhaps examples of nursing notes and discharge summaries lacked efficiency in this task due to a quality deficit, or maybe ChatGPT was not good at classifying notes, meaning our examples often did not correspond to their supposed type.

The second conclusion is that we could try adjusting the prompt to increase the quality of the structure and completeness of the generated notes. However, as our prompt engineering logs [37] show, this had long been attempted, with requests for the notes to be structured like the examples and complete. Once again, due to model limitations, this was to no avail.

Lastly, it was important to remember that the small scale of this evaluation may have had a large influence on these results, meaning we had to be cautious when basing prompt changes on them.

Therefore, based on this analysis, our final decision was to retain the original prompt.

4.3.3 Dataset generation

After finishing the prompt engineering process, we generated the full dataset of 100,000 clinical notes, 25,000 of each type. The generation was performed on the same Virtual Machine as the fine-tuning. It took 17 days, 20 hours, 50 minutes, and 44 seconds to complete.

4.3.4 Dataset post-processing

Finally, we performed rule-based post-processing by automatically eliminating noisy notes through the use of keywords. This was because the model would sometimes act like the prompter and either request the generation of clinical notes or provide instructions on how to generate them. Moreover, on some occasions, it would generate a transcript of the physician's side of a physician-patient interaction. We removed notes with the keywords 'escrever' (*to write*), 'reescrever' (*to rewrite*), 'escreve' (*write*), 'reescreve' (*rewrite*), 'registar' (*to register*), 'estás' (*you are*), 'você' (*you, formal*), 'tu' (*you, informal*). This led to the final dataset.

4.4 Annotation

After generating LusoClin, we opted to provide an additional resource for future research by automatically annotating it at the NER level, leveraging the latest developments in this task.

4.4.1 LLM selection

As seen in the analysis in [Section 2.4.4](#), BERT-based LLMs perform better than generative LLMs in IE tasks, which include NER. Thus, we opted for a BERT-based model, rather than a generative model. Given its characteristics and results, we selected MediAlbertina PT-PT with 1.5 billion parameters and fine-tuning for the NER task [103]. MediAlbertina PT-PT [103] is a BERT-based model that was pre-trained on general domain European Portuguese text [117]. Then, it was further pre-trained on more than 15M sentences and 300M tokens from a dataset of 2.6M medical documents from Portugal’s largest public hospital. This dataset was cleaned by removing nulls and duplicate observations, as well as documents with poor data quality [103]. We selected the version that was fine-tuned for the NER task, which was fine-tuned on 10k annotated medical entities from 4k medical sentences from the same hospital. It is available through the HuggingFace API, under the name ‘medialbertina_pt-pt_1.5b_NER’⁴.

4.4.2 NER Annotation

We annotated LusoClin by performing NER on all notes. The model annotates text with the entity types described in [Table 4.10](#). Descriptions were based on information from the model’s HuggingFace webpage⁴.

Entity type	Description
‘Diagnostico’ (Diagnosis)	Any disease and condition following the ICD-10-CM guidelines.
‘Sintoma’ (Symptom)	Any indication that a patient is experiencing a medical condition.
‘Medicacao’ (Medication/Drug)	Anything that is administrated to the patient, through any route.
‘Dosagem’ (Dosage)	Dosage and frequency of medication administration.
‘Procedimento Medico’ (Medical Procedure)	Anything healthcare professionals do related to patients.
‘Sinal Vital’ (Vital Sign)	Quantifiable and measurable indicators in a patient, always associated with a specific result.
‘Resultado’ (Result)	Results associated with ‘Medical Procedures’ and ‘Vital Signs’. They can be numerical values or descriptors.
‘Progresso’ (Progress)	Anything describing the progress of a patient’s condition.

Table 4.10: MediAlbertina’s entity types.

⁴https://huggingface.co/portugueseNLP/medialbertina_pt-pt_1.5b_NER

We opted to record the annotations in *XML* files, following the structure used by SemClinBr, as shown by the example in [Listing 4.2](#):

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <ANNOTATIONS>
3   <TEXT>O paciente apresentou episódios de dor no peito, queixas de
4   cansaço e fadiga, além de perda de apetite. A avaliação física
5   revelou uma perda de peso significativa. O exame físico não revelou
6   nenhum evidente sinal de doença ou lesão. A avaliação psicológica
7   apontou para uma ansiedade generalizada. O paciente apresentou
8   dificuldade em dormir e sonhos frequentes. A avaliação neurológica
9   não revelou nenhum evidente problema. O paciente declarou ter tido
10  um trauma no passado.</TEXT>
11  <TAGS>
12    <annotation id="1" tag="Sintoma" start="35" end="56" text="dor no peito,
13      queixas"/>
14    <annotation id="2" tag="Sintoma" start="60" end="67" text="cansaço"/>
15    <annotation id="3" tag="Sintoma" start="70" end="76" text="fadiga"/>
16    <annotation id="4" tag="Sintoma" start="86" end="102" text="perda de apetite"/>
17    <annotation id="5" tag="Sintoma" start="135" end="148" text="perda de peso"/>
18    <annotation id="6" tag="Procedimento Medico" start="166" end="178" text="exame
19      físico"/>
20    <annotation id="7" tag="Diagnostico" start="274" end="296" text="ansiedade
21      generalizada"/>
22    <annotation id="8" tag="Sintoma" start="320" end="341" text="dificuldade em
23      dormir"/>
24    <annotation id="9" tag="Sintoma" start="344" end="361" text="sonhos frequentes"
25      />
26    <annotation id="10" tag="Diagnostico" start="457" end="463"
27      text="trauma"/>
28  </TAGS>
29 </ANNOTATIONS>

```

Listing 4.2: Example of annotated clinical note from LusoClin.

The ‘text’ tag includes the unstructured text of the clinical note. The ‘tags’ tag contains the annotations, each represented by an ‘annotation’ tag. In each annotation, ‘tag’ is the entity’s type, ‘start’ and ‘end’ are the indices of the entity’s position in the text, and ‘text’ is the entity as written in the note. The annotation was performed on the same Virtual Machine as the fine-tuning and the generation, and took 2 hours, 47 minutes, and 47 seconds to complete.

Chapter 5

LusoClin evaluation

In this chapter, we describe the human and automatic methods used to evaluate LusoClin. Then, we present the results of these methods. For each result, we analyse what they mean for the dimensions they evaluate. Lastly, we provide a summary of the conclusions drawn from this analysis.

5.1 Evaluation

Post-processing described in [Section 4.3.4](#) eliminated multiple notes from the 100,000 initially generated. Thus, our final version of LusoClin contained 24,759 admission notes, 24,411 ambulatory notes, 24,639 discharge summaries, and 24,762 nursing notes, for a grand total of 98,571 synthetic clinical notes. We performed both types of evaluation on LusoClin’s data: human and automatic. We conducted human evaluation due to its importance in evaluating synthetic data, as existing literature demonstrates. The goal of automatic evaluation methods was complementing the human evaluation and collecting analysis that was able to encapsulate the whole dataset. [Table 5.1](#) describes the evaluations methods used and indicates the dimensions to which they pertained.

5.1.1 Human evaluation

For this evaluation, we had the collaboration of medical professionals and medical students enrolled in the third, fourth, fifth, and sixth years of a six-year medical degree. These professionals and students came from the Faculty of Medicine of the University of Porto and the Abel Salazar School of Medicine and Biomedical Sciences. To publicise the application, we sent out emails to students of these institutions that were enrolled in the noted years. We also had the help of a professor from the Faculty of Medicine of the University of Porto, who publicised it to his students and fellow medical professionals. Evaluations contemplated in this dissertation were conducted from May 15, 2025, to June 2, 2025.

To enable their collaboration, we developed a web application¹, that sought to strike a balance between utility and simplicity of the evaluation process.

¹<http://notasclinicas.inesctec.pt>

Evaluation method	Description	Evaluated dimension(s)
Human evaluation	Turing test by medical students and professionals, where they evaluated notes from LusoClin and SemClinBr as real or synthetic and justified their answer	Utility, fidelity
Perplexity	Average perplexity of two open-source generative LLMs, trained on Portuguese text (Glória [87] and Cabrita [78]), on LusoClin and SemClinBr	Utility, fidelity
Textual and entity statistics	Comparison between LusoClin and SemClinBr on: average number of words and sentences per note, for each note type; average number of entities per note, for each note type; average number of entities per note, for each entity type; frequency of each entity type	Utility, fidelity
t-SNE plot	Comparison between t-SNE plots of subset of LusoClin and SemClinBr	Utility, fidelity, diversity
CMD	Average CMD between subset of LusoClin and SemClinBr	Fidelity
Cosine similarity	Average cosine similarity of every possible pair of samples from subset of LusoClin, compared with average cosine similarity of every possible pair of samples from SemClinBr	Diversity
	Average cosine similarity of every possible pair of one sample from subset of LusoClin and one sample from SemClinBr, of any note type	Privacy
	Average cosine similarity of every possible pair of one sample from subset of LusoClin and one sample from SemClinBr, of the same note type	Privacy
Hausdorff distance	Average Hausdorff distance between pairs of one sample from subset of LusoClin and one sample from SemClinBr, of any note type	Privacy
	Average Hausdorff distance between pairs of one sample from subset of LusoClin and one sample from SemClinBr, of the same note type	Privacy

Table 5.1: Summary of evaluation methods used and dimensions they evaluate.

When accessing the website, users were prompted to create an account, login, or proceed without authentication. The application's menu provided a summary of its goal and access to a page detailing this project and another explaining the application's functionalities.

Upon starting an evaluation, users were presented with a straightforward format. They were shown a clinical note, which had a 90% chance of being a synthetic note from LusoClin and a 10% chance of being a real note from SemClinBr (they would not know which). They had to answer whether they thought the note was real or synthetic, and had the opportunity to offer a justification for their answer, which they could pick from various suggestions or write freely. The goal of suggestions was to speed up and simplify the evaluation process by providing users with expectedly common justifications. These were based on the answers given by the medical students who participated in the small-scale evaluation described in [Section 4.3.2](#), when asked why they thought notes were synthetic. They were also based on common issues we encountered during prompt engineering. Users could also select excerpts from the note and leave comments specifically pertaining to that excerpt. At any point, they could erase their progress in the current evaluation. After finishing, they could save the evaluation and move on to another one, or save and return to the menu. [Figure 5.1](#) shows an instance of the evaluation page with all the described options.

The screenshot displays the 'Nota de Alta' evaluation interface. At the top, a clinical note is presented in a light blue box. Below the note, there are two radio buttons for 'Esta nota é:' with 'Real' selected. A 'Justificação:' text area is provided for user input. Below this, a list of 'Sugestões de justificação:' offers common reasons for a note being synthetic. A 'Comentário:' pop-up window is visible over the note, showing a text input field and a 'Limpar Seleção' button. At the bottom, there are three buttons: 'Limpar', 'Confirmar e Sair', and 'Confirmar e Continuar'. A footer line provides contact information for doubts or suggestions.

Nota de Alta ?

O paciente foi internado após apresentar dor torácica com irradiação para dorso e abdômen em 12/10, de forte intensidade. Foram realizados exames de imagem, incluindo TC de torax e abdômen, que revelaram uma dissecação de aorta descendente a partir da emergência da subclávia esquerda. Apesar de uma avaliação pela cirurgia vascular que considerou o procedimento cirúrgico como contraindicado devido a **complicações já estabelecidas**, o paciente evoluiu para assistolia e óbito em 13/12, após implante de stent na ponte de safena foi realizado sem o uso de ticlopidina e acompanhamento em caso de

Comentário:

'Ex: termo médico inexistente'

Limpar Seleção

Esta nota é:

☒ Real
☐ Sintética

Justificação:

A nota faz sentido e tem uma organização realista.

Sugestões de justificação:

A nota faz sentido e tem uma organização realista.
Informação médica incorreta.
Dados médicos corretos, mas organização atípica da informação.
Estrutura de frases não parece humana.
Dados médicos desconexos.
Escolha de informação a relatar não faz sentido.
Falta informação crucial.
Contém acrônimos inexistentes.

Limpar Confirmar e Sair Confirmar e Continuar

Para dúvidas, questões ou sugestões contacte: up201905189@up.pt | danny.felix16@gmail.com

Figure 5.1: Web application's evaluation page.

To increase the value of this evaluation, we opted to collect more than one evaluation per note.

We settled on two evaluations per note, as any more would greatly reduce the number of different notes evaluated. To achieve two evaluations per note, we used a strategy in which, upon starting an evaluation, users were given a note that was randomly selected from the set of clinical notes that had already been evaluated once, with the exclusion of notes that had been evaluated by that user. If none were available in this set, a random note with no previous evaluations was selected.

Authenticated users also had the opportunity to access their profile page, seen in [Figure 5.2](#), which included some statistics on their evaluations and allowed them to select and review any of their completed evaluations.

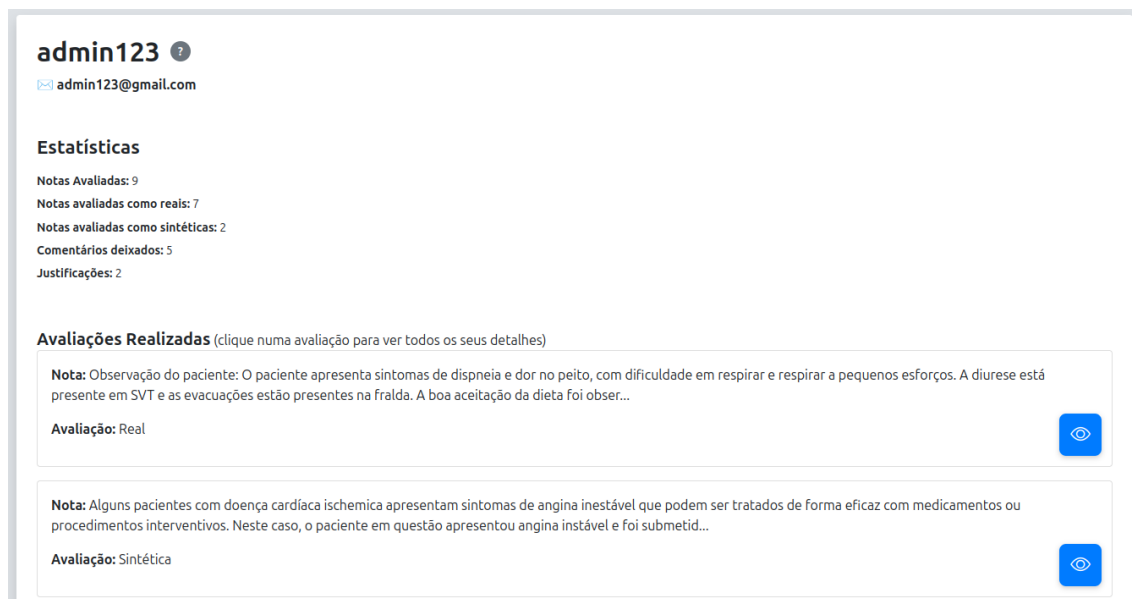


Figure 5.2: Web application’s user profile page.

5.1.2 Automatic evaluation

For the t-SNE plot, CMD, cosine similarity, and Hausdorff distance, we were forced to use a subset of LusoClin due to the infeasibility of the tasks on the whole dataset, as they were too computationally expensive and time-consuming. For that, we decided to match SemClinBr’s size of 1,000 clinical notes (250 of each type), randomly selected. This subset is available in the provided repository [37]. Furthermore, these four methods make use of sentence-level embeddings of the data. For this, we used SentenceBERT embeddings [116], a modification of the pre-trained BERT network that creates sentence-level embeddings where semantically similar sentences are close in vector space. Portuguese is one of the languages supported by the multilingual version of SentenceBERT². For perplexity and textual statistics, the nature of the computations allowed us to perform them on the whole of LusoClin. For perplexity, textual statistics, and cosine similarity, we used SemClinBr as baseline. For t-SNE plot and CMD, as baseline, we would require another study that used these methods between a synthetic Portuguese dataset and a real Portuguese dataset. As there are none, we resorted to using results of an English work [158] as baseline.

²<https://huggingface.co/sentence-transformers/distiluse-base-multilingual-cased-v1>

First, we calculated the average perplexity of two generative models on LusoClin and SemClinBr. The generative models we employed were Glória [87], which is open-source and trained on European Portuguese, and Cabrita [78], which is open-source and trained on Brazilian Portuguese. The goal was to compare whether the models could predict the next tokens in a sequence of LusoClin’s data as easily as in a sequence of SemClinBr’s data.

Next, we calculated various statistics regarding the text and entities of the synthetic and real notes. We calculated the average number of words and sentences per LusoClin note for each note type and compared it with the same numbers for SemClinBr; we computed the frequency of each entity type on LusoClin and compared it with SemClinBr. To have the ability to make a direct comparison, we re-annotated SemClinBr with MediAlbertina PT-PT with 1.5 billion parameters; we calculated the average number of entities per LusoClin note for each note type and compared it with the same numbers for SemClinBr; and we calculated the average number of entities of each entity type per LusoClin note and compared it with the same numbers for SemClinBr.

Afterward, we computed the t-SNE plots of the LusoClin subset and SemClinBr, looking to find similar data distributions. We also calculated the average CMD between the LusoClin subset and SemClinBr to examine whether the distribution of the synthetic samples had shifted away from the clinical domain.

Then, we calculated the average cosine similarity of every sample pair in the LusoClin subset. For this, we took each sentence in a sample, and calculated its similarity with every sentence in the other sample, repeating this for all sentences and calculating the average of these results. After doing this for every pair of samples, we averaged it to get the final result. We performed the same calculation for SemClinBr and compared results to check if LusoClin was as diverse as a real dataset. We used the same process to calculate the average cosine similarity of pairs of one LusoClin and one SemClinBr sample of any note type. Then, we repeated this calculation for synthetic-real pairs of the same note type. We compared these results with the privacy preservation baseline proposed by Hernandez et al. [58].

Lastly, we calculated the average Hausdorff distance between pairs of one synthetic and one real note of any note type, and repeated this for synthetic-real pairs of the same note type. Once again, we compared these results with the privacy preservation baseline proposed by Hernandez et al. [58].

5.2 Results

In this section, we present and analyse the results of the human and automatic evaluations conducted on LusoClin.

5.2.1 Human evaluation

Through our web application, we collected a total of 149 evaluations of 78 different clinical notes. Due to our strategy of two evaluations per note, 71 of these 78 notes had two evaluations. The distribution of evaluations of these 71 notes (57 LusoClin notes and 14 SemClinBr notes) is displayed

in Table 5.2.

Note type	LusoClin notes			SemClinBr notes		
	2 real votes	1 of each	2 synthetic votes	2 real votes	1 of each	2 synthetic votes
Admission notes	0	2	6	0	1	0
Ambulatory notes	1	10	9	1	4	2
Discharge summaries	1	8	8	0	2	0
Nursing notes	1	2	9	1	1	2
Total	3	22	32	2	8	4

Table 5.2: Evaluation distribution of the 71 clinical notes that were evaluated twice.

We opted for Cohen’s Kappa [22] to calculate inter-evaluator agreement. Cohen’s Kappa [22] is a statistical coefficient that measures inter-evaluator reliability for categorical items. It quantifies the degree of agreement between two evaluators who independently classify items into a fixed set of classifications, while adjusting for the agreement that could occur by chance. Unlike simple percent agreement, Cohen’s Kappa provides a more robust assessment of consistency by considering the possibility that evaluators may coincide in their judgments purely due to random chance [93].

Using the values from Table 5.2, we can calculate the Cohen’s Kappa of those evaluations to be -0.0441. This value implies no meaningful agreement and even systematic disagreement beyond chance, suggesting inconsistencies among the evaluators [93, 127, 148]. Such a result may originate from the inherent difficulty of the task, where LusoClin and SemClinBr notes are difficult to distinguish, or from discrepancies in the individual criteria used by evaluators to make their judgments [127].

For the rest of this section about the results of the human evaluation, we take into account solely the 41 clinical notes that had two concordant evaluations, and their respective 82 evaluations.

Table 5.3 summarizes the results of the Turing test for these notes. As we can see, only 9% of LusoClin notes were able to pass the Turing test. However, 67% of SemClinBr notes were evaluated as synthetic. One plausible explanation for this is that evaluators, being aware of the task at hand, likely paid very close attention to every detail in the notes and were stricter in their requirements for realism, causing them to doubt aspects of real notes as well. Therefore, it is important to examine the reasons evaluators provided for why they viewed notes as synthetic.

Note type	LusoClin notes		SemClinBr notes	
	Voted real	Voted synthetic	Voted real	Voted synthetic
Admission notes	0 (0%)	6 (100%)	0 (0%)	0 (0%)
Ambulatory notes	1 (10%)	9 (90%)	1 (33%)	2 (67%)
Discharge summaries	1 (13%)	8 (87%)	0 (0%)	0 (0%)
Nursing notes	1 (10%)	9 (90%)	1 (33%)	2 (67%)
Total	3 (9%)	32 (91%)	2 (33%)	4 (67%)

Table 5.3: Human evaluation - Turing test results.

Many different justifications were given. In Table 5.4, we show the ratios of concordant and discordant justifications. Of these, the notes with the strongest assessments are those with concordant justifications, as it means both evaluators agree on what makes those notes synthetic. SemClinBr had no notes with concordant justifications, while LusoClin had 5. Analysing those notes, we find the following justifications: presence of medical errors for 2 of them, presence of meta-data for 1, absence of key information for 1, and an overall synthetic feel for 1. Listing 5.1 shows the note with meta-data and Listing 5.2 shows one of the notes with medical errors.

	LusoClin notes		SemClinBr notes	
	No. of notes	% of notes	No. of notes	% of notes
Concordant justifications	5	15.6%	0	0.0%
Discordant justifications	8	25.0%	0	0.0%
No justification on one	17	53.1%	2	50.0%
No justification on either	2	6.3%	2	50.0%

Table 5.4: Ratios of concordant and discordant justification for why notes were evaluated as synthetic.

Não incluía nenhuma secção do estilo 'Idade: 52 anos'. Não repita a palavra 'paciente' várias vezes. A tua resposta deve incluir apenas o nota de internamento. O paciente foi encaminhado ao HCPA por dor torácica com suspeita de infarto agudo do miocárdio. A actinoplaquia transcorrente revelou lesões em DAE, colocação de stent com sucesso, e outras lesões em Cx discreta e irregulares, além de ateromatose difusa. Ao longo da internação, o paciente apresentou dor torácica e cianose, s o tratado com analgesias e oxigênio. No entanto, a hipotensão causada pelo uso de beta-bloqueador levou à suspensão temporária do medicamento. O paciente foi monitorado e mantido estável, com controle da dor e melhoria das lesões. A internação foi prolongada devido à complexidade das lesões. O paciente recebeu tratamento com antiagregadores, antiinflamatórios e antiplaca. A recuperação foi lenta, mas estável.

Listing 5.1: Admission note from LusoClin that contains meta-data, according to two evaluators.

Registo de consulta O senhor Lucas, 78 anos, que está a fazer a sua segunda consulta, apresenta referências de incontinência urinária aos esforços, mas nega qualquer queixa urinária. Tem diabetes mellitus (dm) e doença de esclerose lenta progressiva (dslp) há 20 anos. Sua histerectomia foi realizada aos 45 anos e a remoção do ovário direito aos 19 anos. O seu histórico médico inclui a prescrição de vários medicamentos, incluindo alevotiroxina, metformina, omeprazol, losartana, bezafibrato, premarin, citrato de cálcio e colecalciferol. Os resultados dos exames incluem creatinina de 1,4, urina com 68, Cdk epi de 35,9, glicemia de 128, potássio de 5,2, urina ligeiramente turva, pH 7, leucócitos de 29, gases venosos com pH 7,36 e pco2 de 52,60. O paciente apresenta varizes bilaterais mais pronunciadas no abdome. O seu pressão arterial é de 130/80, e o fator de coagulação é de O paciente foi classificado como normocorada e hidratada, com pulmões e coração em ordem.

Listing 5.2: Ambulatory note from LusoClin that contains medical errors, according to two evaluators.

Given that most justifications did not attain a two-evaluator agreement, it is important to also analyse those, even if they do not hold the same strength over their respective notes. In [Table 5.5](#), we display all justifications provided. Due to varying justifications, it is challenging to draw conclusions on general issues with notes from LusoClin. Furthermore, most of these justifications stemmed from only one of the two evaluators of each note, meaning they are not authoritative and should not be used for generalizations about the utility and fidelity of the dataset as a whole. However, analysing justifications by estimating how much they may affect the utility and fidelity of synthetic notes can be valuable in understanding possible issues of LusoClin and potential improvements.

‘Non-sensical’ notes would have no utility or fidelity. Notes with ‘medical errors’ and ‘non-existent medical terminology’ could be expected to have very little utility or fidelity, as the existence of incorrect medical content may pose a significant risk for downstream applications. Notes that are ‘missing key information’ can still have plenty of utility and fidelity, as this justification only reports on clinical content that is missing, which means the content present on the note is likely medically valid. The rest of the justifications suggest their respective notes have advantageous amounts of utility and fidelity, with varying degrees. These justifications point to their notes containing correct and coherent medical information, as issues that make their synthetic nature identifiable are not related to their clinical content.

Regarding SemClinBr notes, no analysis is feasible, as only 2 justifications were given.

We also observe that 13 of the 82 evaluations (15.85%) directly used justifications suggested in the web application.

Justification	LusoClin		SemClinBr	
	N	%	N	%
Feels synthetic	7	17.07%	0	0.0%
Medical errors	7	17.07%	1	50.00%
Missing key information	6	14.63%	0	0.0%
Non-sensical	5	12.20%	0	0.0%
Contains meta-data	4	9.76%	0	0.0%
Repetitive	3	7.32%	0	0.0%
Incorrect structure and/or grammar	3	7.32%	0	0.0%
Unusual terminology	3	7.32%	0	0.0%
Brazilian words	2	4.88%	1	50.00%
Non-existent medical terminology	1	2.44%	0	0.0%

Table 5.5: Summary of justification why notes were evaluated as synthetic. 24 LusoClin and 6 SemClinBr evaluations provided no justification.

Lastly, we have to look at the text excerpts selected and commented on by evaluators, which Table 5.6 summarizes. This feature was used 26 times in 12 synthetic evaluations of 12 different clinical notes. All of these 12 notes were from LusoClin. No clinical note had both of its evaluations contain selected excerpts, so agreement statistics are not applicable. Even though this feature was sparsely used by evaluators, it still offers some insight into issues with notes from LusoClin. The most frequent comment was regarding the presence of Brazilian words, which constituted 30% of commented excerpt selections. Stronger negative features, such as ‘medical error’ and ‘non-existent medical term’, only made up 12.50% of excerpt comments. The rest of the comments called out issues that do not entirely jeopardize the utility and fidelity of their respective notes.

Comment about text excerpt	LusoClin	
	N	%
Brazilian word(s)	7	29.17%
Unusual term	4	16.67%
Feels synthetic	2	8.33%
Missing clinical parameter	2	8.33%
Meta-data	2	8.33%
Repetitive	2	8.33%
Non-existent medical term	2	8.33%
Medical error	1	4.17%
Missing key information	1	4.17%
Spanish word(s)	1	4.17%

Table 5.6: Summary of comments made about selected text excerpts on notes evaluated as synthetic. 2 selected excerpts had no associated comment.

5.2.2 Automatic evaluation

Starting with the average perplexities achieved by SemClinBr and LusoClin, displayed in Table 5.7, we can see that LusoClin performed better on both LLMs, meaning the Portuguese generative models we used had less difficulty in predicting the next tokens in a sequence from the synthetic samples. This is a positive result for LusoClin, as it means its content is more predictable. Generally, the more predictable the clinical content, the more likely it is to be medically valid [9, 77, 95]. Therefore, these results point towards LusoClin having medically valid clinical content, which is a good indicator of utility and fidelity. Remarkably, even on Cabrita, the Brazilian Portuguese model, LusoClin outperformed SemClinBr, which is natively Brazilian.

Dataset	Average perplexity	
	Glória	Cabrita
LusoClin	54.94	24.18
SemClinBr	82.57	37.31

Table 5.7: Perplexity results. Both models were better at predicting the next token in the synthetic data than in the real data.

In Figure 5.3, we can see word count distributions for each note type in LusoClin and SemClinBr. Comparing these values, we can see that all types of synthetic notes have, on average, more words than real notes. However, these differences are relatively small, leading us to conclude that LusoClin notes are similar in length to SemClinBr notes. These values also corroborate our conclusion during prompt engineering that the model mostly ignored requests to limit the size of the generated notes: we had specifically requested, in the prompt, that the notes be 130 words long, and all averages are above 145 words, even with our post-processing trimming operations.

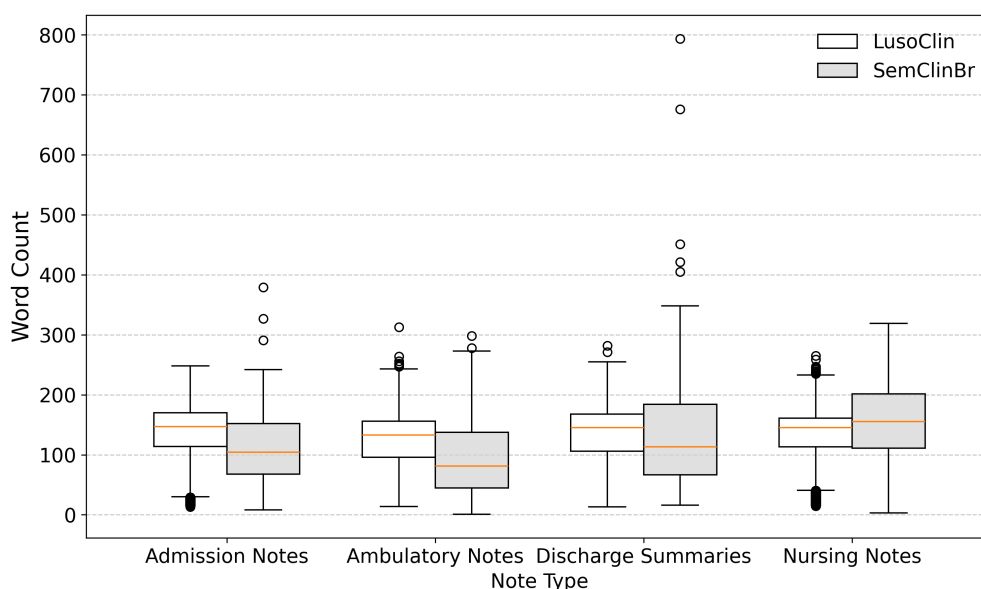


Figure 5.3: Word count distributions for each note type in LusoClin and SemClinBr.

In the case of sentence count distributions, which we can see in [Figure 5.4](#), we can conclude the model was much better at abiding by our specific request of 10 sentences per note, with all note types having an average close to 10 sentences per note.

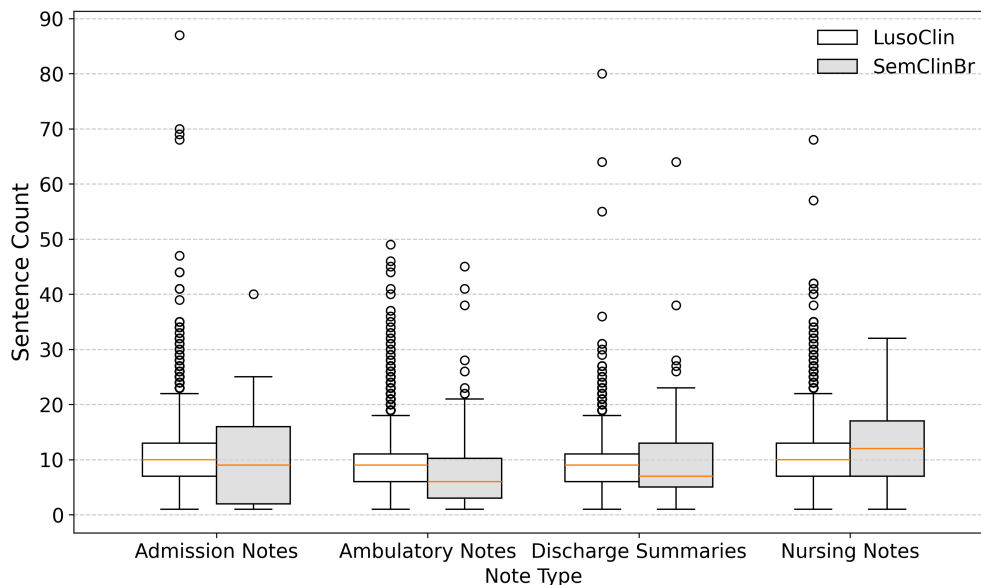


Figure 5.4: Sentence count distributions for each note type in LusoClin and SemClinBr.

[Table 5.8](#) contains the total amount of entities of each entity type for synthetic and real notes, alongside the percentage of the total that that amount represents. The average absolute difference between entity type frequencies is 4.15%, which indicates proximity between the type and amount of entities present in synthetic and real samples. This helps assessing the medical content in LusoClin as realistic, which points towards LusoClin having utility and fidelity.

Entity Type	LusoClin		SemClinBr	
	Total Count	Percentage	Total Count	Percentage
‘Diagnostico’ (Diagnosis)	488,469	26.23%	2,378	16.31%
‘Sintoma’ (Symptom)	315,285	16.93%	1,915	13.14%
‘Procedimento Medico’ (Medical Procedure)	298,557	16.03%	2,228	15.28%
‘Resultado’ (Result)	238,020	12.78%	2,616	17.94%
‘Sinal Vital’ (Vital Sign)	210,827	11.32%	2,617	17.95%
‘Medicacao’ (Medication/Drug)	151,778	8.15%	1,331	9.13%
‘Dosagem’ (Dosagem)	92,746	4.98%	1,281	8.79%
‘Progresso’ (Progress)	66,863	3.59%	212	1.45%

Table 5.8: Frequency of each entity type on LusoClin and SemClinBr.

[Figure 5.5](#) compares the entity count distributions in LusoClin and SemClinBr notes, for each of the four note types. The results indicate a marked disparity, as all types of LusoClin notes have a higher number of entities than their SemClinBr counterparts. This could mean that our generative model created clinical notes that are denser in medical terms.

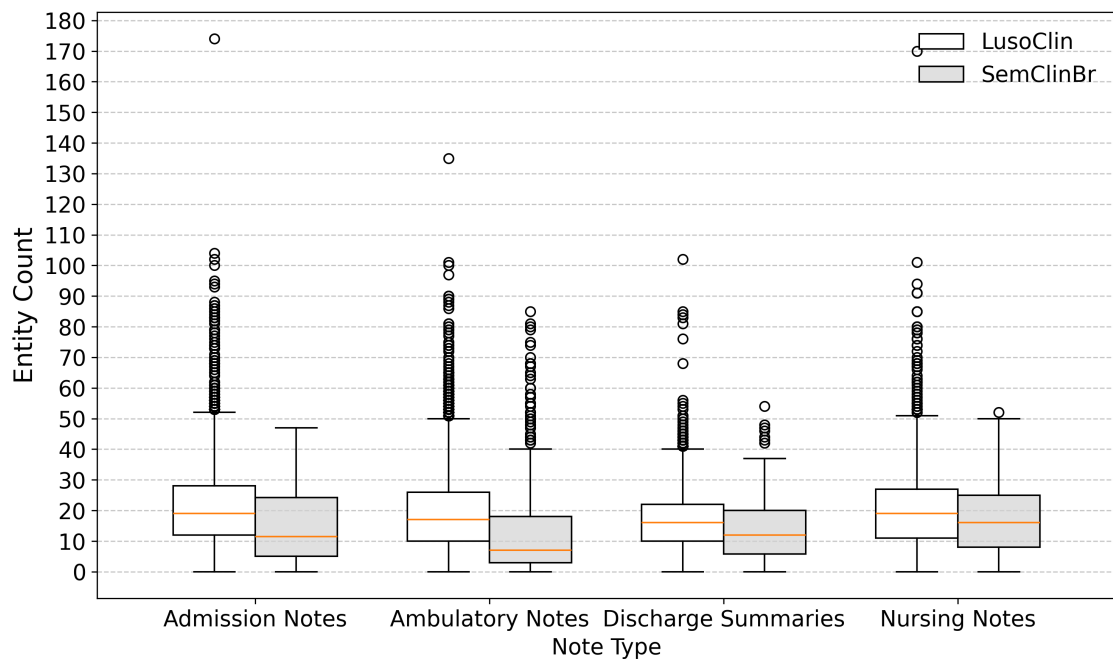


Figure 5.5: Synthetic clinical notes have more entities than real notes.

Figure 5.6 and Figure 5.7 compare the entity count distributions in LusoClin and SemClinBr notes, for each of the eight entity types. We split it into two graphs for easier visualization. These show that LusoClin and SemClinBr have similar distributions of all entity types, except ‘Diagnosis’ and ‘Medical Procedure’ entities, in which LusoClin is denser.

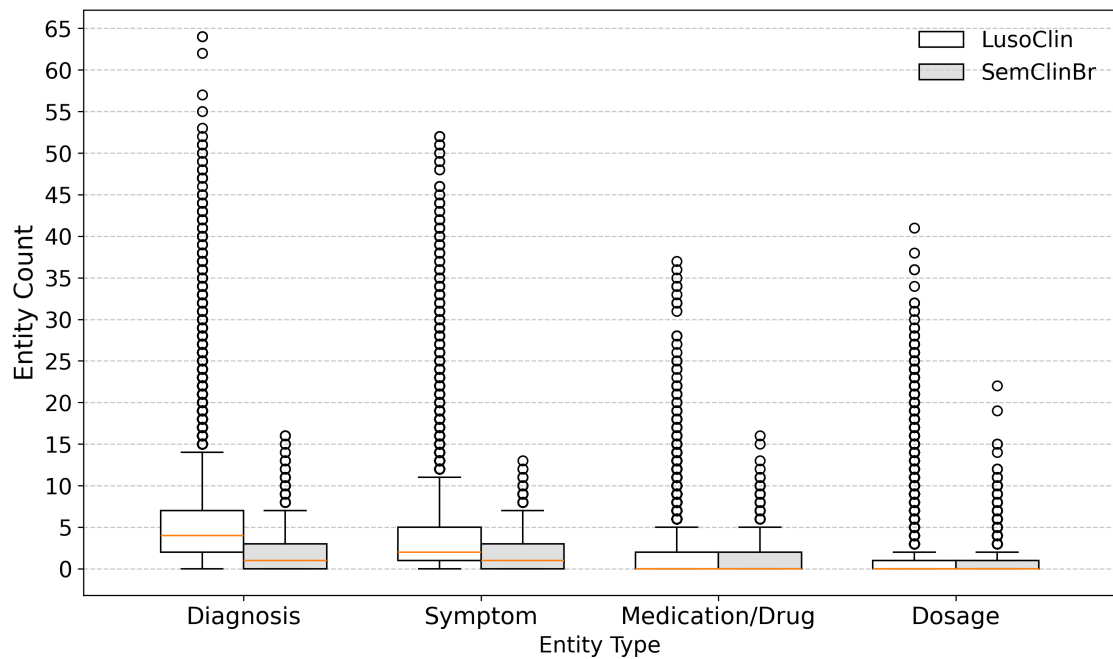


Figure 5.6: Entity count distributions for each entity type in LusoClin and SemClinBr - Part 1.

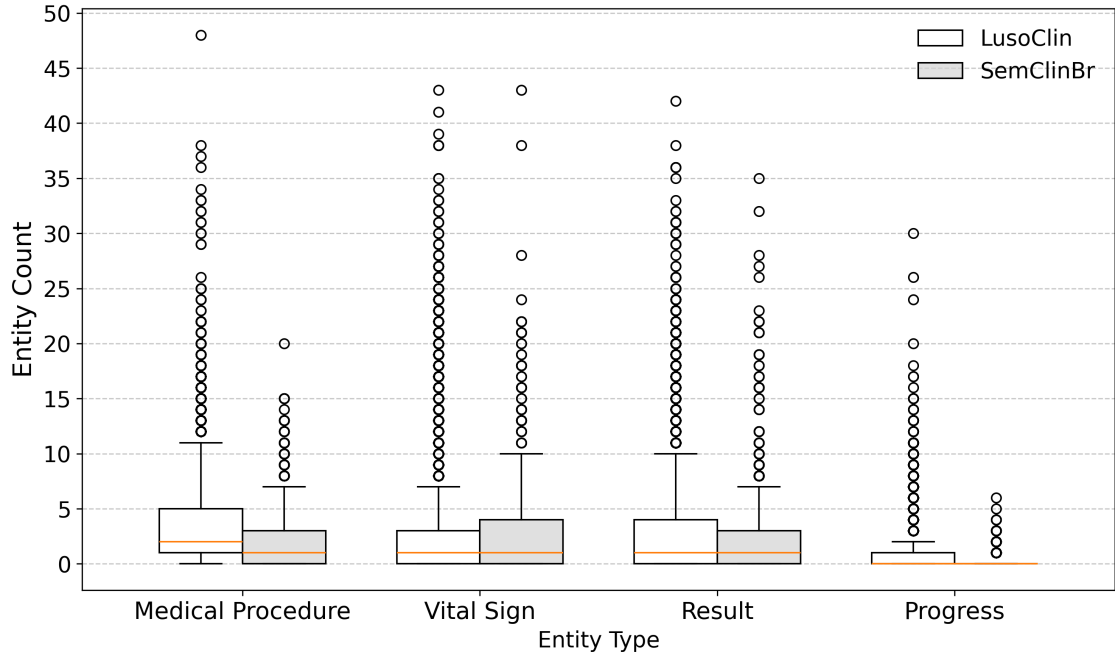


Figure 5.7: Entity count distributions for each entity type in LusoClin and SemClinBr - Part 2.

Regarding the t-SNE plots, seen in Figure 5.8, the LusoClin subset appears to only display some overlap with SemClinBr. This indicates LusoClin’s data distribution should be more similar to SemClinBr’s than it is, for LusoClin to be as useful and faithful as SemClinBr. If we analyse the t-SNE plots in Xu et al.’s work [158], which is our English baseline, it corroborates our observation, as the overlap between their synthetic data distribution and their ground truth is greater than ours. On the other hand, spacing between the data points of LusoClin appears similar to SemClinBr, as the English baseline also suggests [158]. It is an argument towards LusoClin being as diverse as SemClinBr, which is a positive result, as SemClinBr is a multi-specialty dataset.

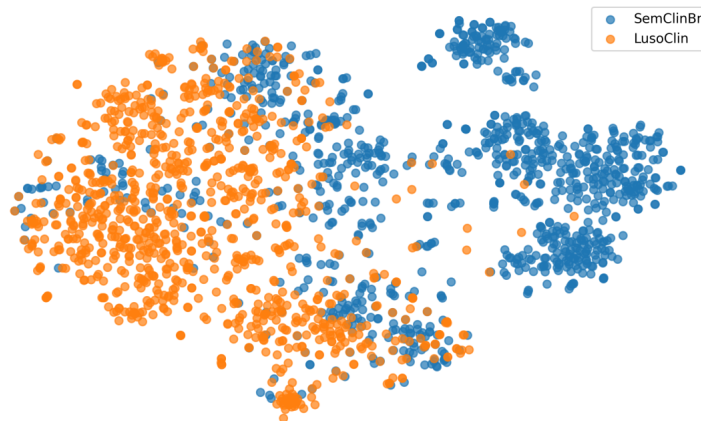


Figure 5.8: t-SNE plots of LusoClin and SemClinBr.

As for CMD, LusoClin obtained a value of 0.413. If we look at our English baseline [158], we can clearly see that LusoClin outperforms the baselines on this particular metric. This low

CMD value means the data distribution of LusoClin did not shift away from SemClinBr’s clinical domain, which is a positive result regarding its fidelity.

Shifting our focus to cosine similarity, the results in Table 5.9 show that the LusoClin subset achieved a lower pairwise cosine similarity than SemClinBr. This is a positive result, as it points towards LusoClin being more diverse than SemClinBr. It could be argued that this might mean LusoClin’s data deviated from the clinical into the general domain, which is naturally more diverse. However, the human evaluation and CMD results contradict this negative conclusion.

Data source	Average cosine similarity
Pairs of LusoClin samples	0.096
Pairs of SemClinBr samples	0.1373

Table 5.9: Average cosine similarity between pairs of samples from the same dataset.

As Table 5.10 demonstrates, the average cosine similarity between datasets was very low. For this calculation, Hernandez et al. [58] propose a threshold of 0.5 for privacy preservation. Since both our type-sensitive and type-insensitive results are significantly lower than the threshold, they constitute a positive indicator of privacy preservation.

Note type	Average cosine similarity
All types	0.0491
Admission notes	0.0541
Ambulatory notes	0.0473
Discharge summaries	0.0703
Nursing notes	0.0682

Table 5.10: Average cosine similarity between pairs of one LusoClin and one SemClinBr note.

In terms of Hausdorff distances, Hernandez et al. [58] propose a threshold of 1 for privacy preservation. Our results, displayed in Table 5.11, show values over 1 for all computed distances. Thus, this is another positive indicator of privacy preservation.

Note type	Average Hausdorff distance
All types	1.2768
Admission notes	1.2748
Ambulatory notes	1.2760
Discharge summaries	1.2757
Nursing notes	1.2517

Table 5.11: Average Hausdorff distance between pairs of one LusoClin and one SemClinBr note.

5.3 Conclusions

After compiling and analysing the results from the various applied evaluation methods, we can draw some conclusions about LusoClin.

Utility and fidelity are the most important dimensions and include the strongest method we applied, which is human evaluation. Human evaluation points to several potential issues with the utility and fidelity of LusoClin, namely non-sensical notes and notes with erroneous medical information. However, most notes with synthetic markers possess correct and coherent medical information, meaning they have plenty of utility and fidelity. A majority of notes from LusoClin were unable to pass as real. However, this must be taken into account alongside the fact that two-thirds of the notes from SemClinBr seemed synthetic to evaluators. Automatic evaluation methods of utility and fidelity provided positive results, with LusoClin matching baselines on t-SNE plot and textual and entity statistics and outperforming them on perplexity and CMD. Therefore, results seem to indicate that LusoClin has reasonable and valuable amounts of utility and fidelity, but also presents issues that need to be corrected, especially the low quality of some medical information and the synthetic feel of most notes. We can also conclude that further human evaluation is required.

In terms of diversity, employed methods show LusoClin being as diverse or slightly more so than SemClinBr. Comparison with other baselines would greatly benefit the evaluation of this dimension.

In terms of privacy, even though our methods point to privacy preservation on behalf of LusoClin, the best assurance of this is the fact that SemClinBr, which was the source of examples in prompts, is a professionally de-identified dataset. Nonetheless, the development of further methods to evaluate this dimension in unstructured text would be a valuable addition to this evaluation.

Chapter 6

Conclusions

In this dissertation, research, planning, and development were carried out to generate and annotate, at the named entity recognition level, a large-scale synthetic dataset of clinical notes in European Portuguese, which we named LusoClin.

In this chapter, we begin by presenting our findings and discussing their relation to our research hypothesis. Then, we point out limitations of this project and finish by providing suggestions for future work.

6.1 Findings

To ascertain if this dissertation’s goal was fulfilled, it is imperative that we analyse our results in light of the research hypothesis proposed in [Section 4.1](#).

RH: An open-source LLM can generate a large-scale European Portuguese dataset of synthetic clinical notes with enough utility, fidelity, diversity, and privacy to replace real data in downstream applications.

Our conclusion is that we cannot state our RH has been definitively proven by this dissertation project, since LusoClin has enough diversity and privacy to replace real data in downstream applications, but not enough utility and fidelity.

While we can conclude LusoClin has a reasonable amount of useful and faithful content, it does not have enough to fully replace real data in downstream applications. Automatic evaluation results were overwhelmingly positive in utility and fidelity. However, human evaluation results clearly demonstrate issues that may encompass a considerable portion of LusoClin, namely non-sensical content and medical errors, which entirely invalidate the value of notes with those issues. It also showed some different drawbacks with other notes, although notes with those problems still retained a good amount of utility and fidelity. Human evaluation leads us to state that there is the potential that a sizeable portion of LusoClin has little to no utility or fidelity, while the rest has varying degrees of these dimensions. Furthermore, we can expect around 20% of LusoClin to be able to replace real data. However, the size of the human evaluation leads us to conclude that further human evaluation is required. Moreover, the fact that two-thirds of SemClinBr notes were

evaluated as synthetic demonstrates that human evaluation of synthetic content always has a certain level of contrivance. Evaluators are aware of the existence of synthetic content and specifically look for synthetic markers, often causing them to tighten their requirements for realism.

Our evaluation methods show that LusoClin’s diversity is on par with SemClinBr’s diversity. Given that SemClinBr is a real dataset that is used in downstream applications, we can conclude that LusoClin meets diversity standards to replace real data in these applications.

Privacy preservation is also guaranteed by LusoClin. Our evaluation showed positive results in LusoClin’s privacy preservation. We recognize that available methods, by themselves, do not allow the definitive evaluation of privacy preservation in synthetic unstructured clinical textual data. However, these methods, combined with the fact that SemClinBr, used during fine-tuning and prompt engineering, is a professionally de-identified dataset, strongly indicate that LusoClin has no privacy issues.

Through the analysis in this document, some improvements are clear for future investigation that aims to authoritatively prove this RH. First, an LLM that is better at generating European Portuguese text is required; second, it would be ideal if the employed LLM was more reactive to specific requests within the prompt and was able to successfully handle more instructions without becoming overwhelmed; third, a better system to automatically eliminate noisy content would trim much of the information with no value; lastly, a larger and more detailed human evaluation would allow for stronger conclusions in terms of utility and fidelity.

In conclusion, we believe that LusoClin is valuable in terms of all four evaluated dimensions, but would need more utility and fidelity to fully replace real data in downstream applications, which means there is plenty of room for improvement. [Table 6.1](#) summarizes the results of the evaluation, divided by dimension.

6.2 Project limitations

The hardware available for the development of this project limited the selection of the generative model and the models used to measure perplexity. Due to limited memory, we were unable to use the Llama 3 variants with 8 billion and 70 billion parameters, having to settle for the variant with 3 billion parameters. Additionally, the available memory also prevented us from training all the model’s weights, leading to the use of the QLoRA fine-tuning method. As for the models used to evaluate perplexity, memory issues excluded Sabiá [111] as an option for the Brazilian Portuguese LLM.

The datasets used to fine-tune the model are very small in scale, which calls into question their ability to influence the model through fine-tuning. The European Portuguese dataset, CLINpt, is the smaller of the two, and is composed of single-domain clinical cases, not clinical notes from various domains. The most influential dataset, SemClinBr, which is bigger and used for examples during prompting, is Brazilian Portuguese and originates from Brazil, which may lead to issues both due to the dialect and the populational characteristics of the subjects discussed in its clinical notes.

Dimension	Evaluation method	Results
Utility + fidelity	Human evaluation	Most LusoClin notes identified as synthetic, but with valuable clinical content. Some non-sensical content and medical errors. Majority of SemClinBr notes incorrectly evaluated as synthetic.
	Perplexity	2 LLMs had lower perplexity on LusoClin than SemClinBr.
	Textual and entity statistics	Similar, LusoClin has more entities per note than SemClinBr.
	t-SNE plot comparison	LusoClin should have more similar data distribution to SemClinBr.
Fidelity	CMD between LusoClin and SemClinBr	Low, LusoClin did not shift away from clinical domain.
Diversity	t-SNE plot comparison	LusoClin has similar spacing between data points as SemClinBr.
	Cosine similarity within dataset	LusoClin has lower similarity than SemClinBr.
Privacy	Cosine similarity between LusoClin and SemClinBr	Better than baseline.
	Hausdorff distance between LusoClin and SemClinBr	Better than baseline.

Table 6.1: Summary of evaluation results, divided by dimension.

The human evaluation we conducted was small in comparison to the size of LusoClin. This removes value from generalized conclusions derived from this evaluation. In an attempt to keep the process simple, which increased the number of evaluated notes, we refrained from requesting detailed analyses of evaluated notes. A more thorough evaluation could provide valuable insight into specific problems with LusoClin’s notes and how they could be fixed. Moreover, a portion of evaluators were medical students, not fully formed medical professionals, which may have translated to slightly incorrect judgments about some of the medical circumstances presented in the notes.

Lastly, we did not thoroughly engineer hyperparameters when fine-tuning our chosen LLM.

6.3 Future work

Several avenues may be followed to continue this investigation. If generating synthetic data again is an option, different LLMs could be used, namely the latest Llama 3.3 model with 70 billion parameters, of which an instruction-tuned version exists¹, or the GlóRIA model, which requires instruction-tuning to be able to fulfill this task. Furthermore, a specifically European Portuguese dataset of real clinical notes would be invaluable to be used for fine-tuning and examples during prompt engineering in a project such as this. For future work using LusoClin, the most important

¹<https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct>

avenues would be: increasing the size of the human evaluation; making its questions more thorough and detail-oriented, such as asking evaluators to rate the realism of the notes on a given scale, or requesting they evaluate the notes on specific parameters, like the ones we used in [Section 4.3.2](#); and restricting the role of evaluator to clinical experts. Other important work would be improving the system that automatically identifies and eliminates noise from the dataset and evaluating LusoClin by performing TSTR: training a general-domain generative LLM using LusoClin and assessing its performance on clinical tasks.

References

- [1] Electronic health records. <https://www.cms.gov/priorities/key-initiatives/e-health/records>. Accessed: 2024-10-31.
- [2] Introducing chatgpt. <https://openai.com/index/chatgpt/>. Accessed: 2024-09-04.
- [3] Perplexity of fixed-length models. <https://huggingface.co/docs/transformers/en/perplexity>. Accessed: 2024-10-07.
- [4] United nations - the 17 goals. <https://sdgs.un.org/goals>. Accessed: 2025-06-03.
- [5] What are large language models (llms)? <https://www.ibm.com/topics/large-language-models>. Accessed: 2024-08-25.
- [6] What is synthetic data? <https://research.ibm.com/blog/what-is-synthetic-data>. Accessed: 2024-08-27.
- [7] *SINAPSE, volume 17*. Sociedade Portuguesa de Neurologia, 2017. <https://www.spneurologia.com/publicacoes/sinapse/ano/2017>.
- [8] Kiran Adnan and Rehan Akbar. An analytical study of information extraction from unstructured and multidimensional big data. *Journal of Big Data*, 6(1):91, Oct 2019. doi: 10.1186/s40537-019-0254-8.
- [9] Ali Amin-Nejad, Julia Ive, and Sumithra Velupillai. Exploring transformer text generation for medical dataset augmentation. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4699–4708, Marseille, France, May 2020. European Language Resources Association.
- [10] André Bauer, Simon Trapp, Michael Stenger, Robert Leppich, Samuel Kounev, Mark Leznik, Kyle Chard, and Ian Foster. Comprehensive exploration of synthetic data generation: A survey, 2024. doi: 10.48550/arXiv.2401.02524.
- [11] Brett K. Beaulieu-Jones, Zhiwei Steven Wu, Chris Williams, Ran Lee, Sanjeev P. Bhavnani, James Brian Byrd, and Casey S. Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *Circulation: Cardiovascular Quality and Outcomes*, 12(7):e005122, 2019. doi: 10.1161/CIRCOUTCOMES.118.005122.

- [12] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022. doi: 10.48550/arXiv.2204.06745.
- [13] Sid Black, Gao Leo, Phil Wang, Connor Leahy, and Stella Biderman. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow, March 2021. doi: 10.5281/zenodo.5297715.
- [14] Emmanuella Budu, Kobra Etminani, Amira Soliman, and Thorsteinn Rögnvaldsson. Evaluation of synthetic electronic health records: A systematic review and experimental assessment. *Neurocomputing*, 603:128253, 2024. doi: 10.1016/j.neucom.2024.128253.
- [15] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. Legal-bert: The muppets straight out of law school, 2020. doi: 10.48550/arXiv.2010.02559.
- [16] Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3), mar 2024. doi: 10.1145/3641289.
- [17] Dong Chen, Xinda Qi, Yu Zheng, Yuzhen Lu, Yanbo Huang, and Zhaojian Li. Synthetic data augmentation by diffusion probabilistic models to enhance weed recognition. *Computers and Electronics in Agriculture*, 216:108517, 2024. doi: 10.1016/j.compag.2023.108517.
- [18] Lihu Chen, Gaël Varoquaux, and Fabian M. Suchanek. A lightweight neural model for biomedical entity linking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12657–12665, May 2021. doi: 10.1609/aaai.v35i14.17499.
- [19] Kyunghyun Cho, Bart van Merriënboer, Çağlar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014. doi: 10.48550/arXiv.1406.1078.
- [20] Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F. Stewart, and Jimeng Sun. Generating multi-label discrete patient records using generative adversarial networks. In Finale Doshi-Velez, Jim Fackler, David Kale, Rajesh Ranganath, Byron Wallace, and Jenna Wiens, editors, *Proceedings of the 2nd Machine Learning for Healthcare Conference*, volume 68 of *Proceedings of Machine Learning Research*, pages 286–305. PMLR, 18–19 Aug 2017. <https://proceedings.mlr.press/v68/choi17a.html>.
- [21] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. https://proceedings.neurips.cc/paper_files/paper/2015/file/1068c6e4c8051cfd4e9ea8072e3189e2-Paper.pdf.
- [22] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960. doi: 10.1177/001316446002000104.

- [23] Martin R. Cowie, Juuso I. Blomster, Lesley H. Curtis, Sylvie Duclaux, Ian Ford, Fleur Fritz, Samantha Goldman, Salim Janmohamed, Jörg Kreuzer, Mark Leenay, Alexander Michel, Seleen Ong, Jill P. Pell, Mary Ross Southworth, Wendy Gattis Stough, Martin Thoenes, Faiez Zannad, and Andrew Zalewski. Electronic health records to facilitate clinical research. *Clinical Research in Cardiology*, 106(1):1–9, Jan 2017. doi: 10.1007/s00392-016-1025-6.
- [24] Saloni Dash, Andrew Yale, Isabelle Guyon, and Kristin P. Bennett. Medical time-series data generation using generative adversarial networks. In Martin Michalowski and Robert Moskovitch, editors, *Artificial Intelligence in Medicine*, pages 382–391, Cham, 2020. Springer International Publishing. doi: 10.1007/978-3-030-59137-3_34.
- [25] Spiros C. Denaxas and Katherine I. Morley. Big biomedical data and cardiovascular disease research: opportunities and challenges. *European Heart Journal - Quality of Care and Clinical Outcomes*, 1(1):9–16, 06 2015. doi: 10.1093/ehjqcc/qcv005.
- [26] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc., 2023. https://proceedings.neurips.cc/paper_files/paper/2023/file/1feb87871436031bdc0f2beaa62a049b-Paper-Conference.pdf.
- [27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. doi: 10.48550/arXiv.1810.04805.
- [28] Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Shafiq Joty, Boyang Li, and Lidong Bing. Is gpt-3 a good data annotator?, 2023. doi: 10.48550/arXiv.2212.10450.
- [29] Peter Eigenschink, Thomas Reutterer, Stefan Vamosi, Ralf Vamosi, Chang Sun, and Klaudius Kalcher. Deep generative models for synthetic data: A survey. *IEEE Access*, 11:47304–47320, 2023. doi: 10.1109/ACCESS.2023.3275134.
- [30] Matúš Falis, Aryo Pradipta Gema, Hang Dong, Luke Daines, Siddharth Basetti, Michael Holder, Rose S Penfold, Alexandra Birch, and Beatrice Alex. Can GPT-3.5 generate and code discharge summaries? *Journal of the American Medical Informatics Association*, 31(10):2284–2293, 09 2024. doi: 10.1093/jamia/ocae132.
- [31] Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, Jun 2024. doi: 10.1038/s41586-024-07421-0.
- [32] Alvaro Figueira and Bruno Vaz. Survey on synthetic data generation, evaluation methods and gans. *Mathematics*, 10(15), 2022. doi: 10.3390/math10152733.
- [33] Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30(4):681–694, Dec 2020. doi: 10.1007/s11023-020-09548-1.
- [34] David Fraile Navarro, Kiran Ijaz, Dana Rezazadegan, Hania Rahimi-Ardabili, Mark Dras, Enrico Coiera, and Shlomo Berkovsky. Clinical named entity recognition and

- relation extraction using natural language processing of medical free text: A systematic review. *International Journal of Medical Informatics*, 177:105122, 2023. doi: 10.1016/j.ijmedinf.2023.105122.
- [35] Johann Frei and Frank Kramer. Annotated dataset creation through large language models for non-english medical nlp. *Journal of Biomedical Informatics*, 145:104478, 2023. doi: 10.1016/j.jbi.2023.104478.
- [36] Evan French and Bridget T. McInnes. An overview of biomedical entity linking throughout the years. *Journal of Biomedical Informatics*, 137:104252, 2023. doi: 10.1016/j.jbi.2022.104252.
- [37] D. Félix and C. Teixeira Lopes. Dataset of synthetic clinical notes in european portuguese generated using an open-source large language model, along with prompting and evaluation data. INESC TEC, 2025. doi: 10.25747/4GC6-DK48.
- [38] Pedro Henrique Paiola Gabriel Lino Garcia, , and João Paulo Papa. Chatbode, 2024. doi: 10.57967/hf/3317.
- [39] Chang Gao, Wenxuan Zhang, Wai Lam, and Lidong Bing. Easy-to-hard learning for information extraction, 2023. doi: 10.48550/arXiv.2305.09193.
- [40] Fabio Garcea, Alessio Serra, Fabrizio Lamberti, and Lia Morra. Data augmentation for medical imaging: A systematic literature review. *Computers in Biology and Medicine*, 152:106391, 2023. doi: 10.1016/j.combiomed.2022.106391.
- [41] Gabriel Lino Garcia, Pedro Henrique Paiola, Luis Henrique Morelli, Giovani Candido, Arnaldo Cândido Júnior, Danilo Samuel Jodas, Luis C. S. Afonso, Ivan Rizzo Guilherme, Bruno Elias Penteado, and João Paulo Papa. Introducing bode: A fine-tuned large language model for portuguese prompt-based task, 2024.
- [42] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. Chatgpt outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120, 2023. doi: 10.1073/pnas.2305016120.
- [43] Louie Giray. Prompt engineering with chatgpt: A guide for academic writers. *Annals of Biomedical Engineering*, 51(12):2629–2633, Dec 2023. doi: 10.1007/s10439-023-03272-4.
- [44] Andre Goncalves, Priyadip Ray, Braden Soper, Jennifer Stevens, Linda Coyle, and Ana Paula Sales. Generation and evaluation of synthetic patient data. *BMC Medical Research Methodology*, 20(1):108, May 2020. doi: 10.1186/s12874-020-00977-1.
- [45] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, October 2020. doi: 10.1145/3422622.
- [46] Mandeep Goyal and Qusay H. Mahmoud. A systematic review of synthetic data generation techniques using generative ai. *Electronics*, 13(17), 2024. doi: 10.3390/electronics13173509.
- [47] Ralph Grishman. Information extraction. *IEEE Intelligent Systems*, 30(5):8–15, 2015. doi: 10.1109/MIS.2015.68.

- [48] Tracy D Gunter and Nicolas P Terry. The emergence of national electronic health record architectures in the united states and australia: models, costs, and questions. *J. Med. Internet Res.*, 7(1):e3, March 2005. <https://pmc.ncbi.nlm.nih.gov/articles/PMC1550638/>.
- [49] Erkam Guresen and Gulgun Kayakutlu. Definition of artificial neural networks with comparison to other networks. *Procedia Computer Science*, 3:426–433, 2011. doi: 10.1016/j.procs.2010.12.071.
- [50] Muhammad Usman Hadi, qasem al tashi, Rizwan Qureshi, Abbas Shah, amgad muneer, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, and Seyedali Mirjalili. A survey on large language models: Applications, challenges, limitations, and practical usage. July 2023. doi: 10.36227/techrxiv.23589741.v1.
- [51] Muhammad Usman Hadi, Qasem Al Tashi, Abbas Shah, Rizwan Qureshi, Amgad Muneer, Muhammad Irfan, Anas Zafar, Muhammad Bilal Shaikh, Naveed Akhtar, Jia Wu, Seyedali Mirjalili, and Mubarak Shah. Large language models: A comprehensive survey of its applications, challenges, limitations, and future prospects. August 2024. doi: 10.36227/techrxiv.23589741.v6.
- [52] Desta Haileselassie Hagos, Rick Battle, and Danda B. Rawat. Recent advances in generative ai and large language models: Current status, challenges, and perspectives. *IEEE Transactions on Artificial Intelligence*, pages 1–21, 2024. doi: 10.1109/TAI.2024.3444742.
- [53] Zeyu Han, Yuhan Wang, Luping Zhou, Peng Wang, Binyu Yan, Jiliu Zhou, Yan Wang, and Dinggang Shen. Contrastive diffusion model with auxiliary guidance for coarse-to-fine pet reconstruction. In Hayit Greenspan, Anant Madabhushi, Parvin Mousavi, Septimiu Salcudean, James Duncan, Tanveer Syeda-Mahmood, and Russell Taylor, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, pages 239–249, Cham, 2023. Springer Nature Switzerland. doi: 10.1007/978-3-031-43999-5_23.
- [54] Shuang Hao, Wenfeng Han, Tao Jiang, Yiping Li, Haonan Wu, Chunlin Zhong, Zhangjun Zhou, and He Tang. Synthetic data in ai: Challenges, applications, and ethical implications, 2024. doi: 10.48550/arXiv.2401.01629.
- [55] Ayoub Harnoune, Maryem Rhanoui, Mounia Mikram, Siham Yousfi, Zineb Elkaimbillah, and Bouchra El Asri. Bert based clinical knowledge extraction for biomedical knowledge graph construction and analysis. *Computer Methods and Programs in Biomedicine Update*, 1:100042, 2021. doi: 10.1016/j.cmpbup.2021.100042.
- [56] Conor Hassan, Robert Salomone, and Kerrie Mengersen. Deep generative models, synthetic tabular data, and differential privacy: An overview and synthesis, 2023. doi: 10.48550/arXiv.2307.15424.
- [57] S.S. Haykin. *Neural Networks: A Comprehensive Foundation*. International edition. Prentice Hall, 1999. <https://books.google.pt/books?id=bX4pAQAAMAAJ>.
- [58] Gorka; Alberdi Ane; Cilla Rodrigo; Rankin Debbie Hernandez, Mikel; Epelde. Synthetic tabular data evaluation in the health domain covering resemblance, utility, and privacy dimensions. *Methods Inf Med*, 62(S 01):e19–e38, Jan 2023. doi: 10.1055/s-0042-1760247.
- [59] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

- [60] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. doi: 10.48550/arXiv.2106.09685.
- [61] Zhentao Hu, Wei Hou, and Xianxing Liu. Deep learning for named entity recognition: a survey. *Neural Computing and Applications*, 36(16):8995–9022, Jun 2024. doi: 10.1007/s00521-024-09646-6.
- [62] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission, 2020. doi: 10.48550/arXiv.1904.05342.
- [63] Syed Mosaddik Hossain Ifty, Farhana Irin, Md Shihab Sadik Shovon, Mohammad Hamid Hasan Amjad, Proshanta Kumar Bhowmik, Raju Ahmed, Md Rahatul Ashakin, Bayazid Hossain, Mushfiq, Abdus Sattar, Redoyan Chowdhury, and Atiqur Rahman Sunny. Advancements, applications, and future directions of artificial intelligence in healthcare. *Journal of Angiotherapy*, 8(8), Aug 2024. doi:10.25163/angiotherapy.889843.
- [64] Basra Jehangir, Saravanan Radhakrishnan, and Rahul Agarwal. A survey on named entity recognition —datasets, tools, and methodologies. *Natural Language Processing Journal*, 3:100017, 2023. doi: 10.1016/j.nlp.2023.100017.
- [65] Zongcheng Ji, Qiang Wei, and Hua Xu. BERT-based ranking for biomedical entity normalization. In *AMIA Joint Summits on Translational Science*, page 269–277, 2020. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7233044/>.
- [66] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, L  lio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timoth  e Lacroix, and William El Sayed. Mistral 7b, 2023. doi: 10.48550/arXiv.2310.06825.
- [67] Jamin Rahman Jim, Md Apon Riaz Talukder, Partha Malakar, Md Mohsin Kabir, Kamrudin Nur, and M.F. Mridha. Recent advancements and challenges of nlp-based sentiment analysis: A state-of-the-art review. *Natural Language Processing Journal*, 6:100059, 2024. doi: 10.1016/j.nlp.2024.100059.
- [68] Alistair E.W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1):160035, May 2016. doi: 10.1038/sdata.2016.35.
- [69] Indu Joshi, Marcel Grimmer, Christian Rathgeb, Christoph Busch, Francois Bremond, and Antitza Dantcheva. Synthetic data in human analysis: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7):4957–4976, 2024. doi: 10.1109/T-PAMI.2024.3362821.
- [70] Vipina K Keloth, Yan Hu, Qianqian Xie, Xueqing Peng, Yan Wang, Andrew Zheng, Melih Selek, Kalpana Raja, Chih Hsuan Wei, Qiao Jin, Zhiyong Lu, Qingyu Chen, and Hua Xu. Advancing entity recognition in biomedicine via instruction tuning of large language models. *Bioinformatics*, 40(4):btac163, 03 2024. doi: 10.1093/bioinformatics/btac163.

- [71] Imed Keraghel, Stanislas Morbieu, and Mohamed Nadif. Recent advances in named entity recognition: A comprehensive survey and comparative study, 2024. doi: 10.48550/arXiv.2401.10825.
- [72] Shweta Kharya, Sunita Soni, and Tripti Swarnkar. Generation of synthetic datasets using weighted bayesian association rules in clinical world. *International Journal of Information Technology*, 14(6):3245–3251, Oct 2022. doi: 10.1007/s41870-022-01081-x.
- [73] Takoua Khorchani, Yojana Gadiya, Gesa Witt, Delia Lanzillotta, Carsten Claussen, and Andrea Zaliani. Sasc: A simple approach to synthetic cohorts for generating longitudinal observational patient cohorts from covid-19 clinical data. *Patterns*, 3(4), 2022. doi: 10.1016/j.patter.2022.100453.
- [74] Diksha Khurana, Aditya Koli, Kiran Khatter, and Sukhdev Singh. Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3):3713–3744, Jan 2023. doi: 10.1007/s11042-022-13428-4.
- [75] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013. <http://web2.cs.columbia.edu/~blei/fogm/2018F/materials/KingmaWelling2013.pdf>.
- [76] Gleb Kumichev, Pavel Blinov, Yulia Kuzkina, Vasily Goncharov, Galina Zubkova, Nikolai Zenovkin, Aleksei Goncharov, and Andrey Savchenko. Medsyn: Llm-based synthetic medical text generation framework. In Albert Bifet, Tomas Krilavičius, Ioanna Miliou, and Sławomir Nowaczyk, editors, *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track*, pages 215–230, Cham, 2024. Springer Nature Switzerland. doi: 10.1007/978-3-031-70381-2_14.
- [77] Sunjun Kweon, Junu Kim, Jiyoung Kim, Sujeong Im, Eunbyeol Cho, Seongsu Bae, Jungwoo Oh, Gyubok Lee, Jong Hak Moon, Seng Chan You, Seungjin Baek, Chang Hoon Han, Yoon Bin Jung, Yohan Jo, and Edward Choi. Publicly shareable clinical large language model built on synthetic clinical notes, 2024. doi: 10.48550/arXiv.2309.00237.
- [78] Celio Larcher, Marcos Piau, Paulo Finardi, Pedro Gengo, Piero Esposito, and Vinicius Caridá. Cabrita: closing the gap for foreign languages, 2023.
- [79] Atif Latif and Jihie Kim. Evaluation and analysis of large language models for clinical text augmentation and generation. *IEEE Access*, 12:48987–48996, 2024. doi: 10.1109/ACCESS.2024.3384496.
- [80] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 09 2019. doi: 10.1093/bioinformatics/btz682.
- [81] Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, R. Andrew Taylor, Harlan M. Krumholz, and Dragomir Radev. Neural natural language processing for unstructured data in electronic health records: A review. *Computer Science Review*, 46:100511, 2022. doi: 10.1016/j.cosrev.2022.100511.

- [82] Zhijing Li, Chen Li, Yu Long, and Xuan Wang. A system for automatically extracting clinical events with temporal information. *BMC Medical Informatics and Decision Making*, 20(1):198, Aug 2020. doi: 10.1186/s12911-020-01208-9.
- [83] Nut Limsopatham and Nigel Collier. Normalising medical concepts in social media texts by learning semantic representation. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1014–1023, 2016. <https://aclanthology.org/P16-1096.pdf>.
- [84] Xiong Liu, Greg L. Hersch, Iya Khalil, and Murthy Devarakonda. Clinical trial information extraction with bert, 2021. doi: 10.48550/arXiv.2110.10027.
- [85] Lin Long, Rui Wang, Ruixuan Xiao, Junbo Zhao, Xiao Ding, Gang Chen, and Haobo Wang. On llms-driven synthetic data generation, curation, and evaluation: A survey, 2024. doi: 10.48550/arXiv.2406.15126.
- [86] Fábio Lopes, César Teixeira, and Hugo Gonalo Oliveira. Contributions to clinical named entity recognition in Portuguese. In Dina Demner-Fushman, Kevin Bretonnel Cohen, Sophia Ananiadou, and Junichi Tsujii, editors, *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 223–233, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5024.
- [87] Ricardo Lopes, Joao Magalhaes, and David Semedo. GlóRIA: A generative and open large language model for Portuguese. In Pablo Gamallo, Daniela Claro, António Teixeira, Livy Real, Marcos Garcia, Hugo Gonalo Oliveira, and Raquel Amaro, editors, *Proceedings of the 16th International Conference on Computational Processing of Portuguese*, pages 441–453, Santiago de Compostela, Galicia/Spain, March 2024. Association for Computational Linguistics.
- [88] Konstantin Lopyrev. Generating news headlines with recurrent neural networks. *CoRR*, abs/1512.01712, 2015. doi: 10.48550/arXiv.1512.01712.
- [89] Isabelle Lorge, Dan W. Joyce, Niall Taylor, Alejo Nevado-Holgado, Andrea Cipriani, and Andrey Kormilitzin. Detecting the clinical features of difficult-to-treat depression using synthetic data from large language models, 2024. doi: 10.48550/arXiv.2402.07645.
- [90] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. doi: 10.48550/arXiv.1711.05101.
- [91] Xuezhe Ma and Eduard H. Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *CoRR*, abs/1603.01354, 2016. doi: 10.48550/arXiv.1603.01354.
- [92] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. <https://www.cambridge.org/us/universitypress/subjects/computer-science/knowledge-management-databases-and-data-mining/introduction-information-retrieval?format=HB&isbn=9780521865715>.
- [93] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochem Med (Zagreb)*, 22(3):276–282, 2012. doi: 10.1177/001316446002000104.
- [94] Tahir Mehmood, Ivan Serina, Alberto Lavelli, Luca Putelli, and Alfonso Gerevini. On the use of knowledge transfer techniques for biomedical named entity recognition. *Future Internet*, 15(2), 2023. doi: 10.3390/fi15020079.

- [95] Oren Melamud and Chaitanya Shivade. Towards automatic generation of shareable synthetic clinical notes using neural language models, 2019. doi: 10.48550/arXiv.1905.07002.
- [96] Pietro Melzi, Ruben Tolosana, Ruben Vera-Rodriguez, Minchul Kim, Christian Rathgeb, Xiaoming Liu, Ivan DeAndres-Tame, Aythami Morales, Julian Fierrez, Javier Ortega-Garcia, Weisong Zhao, Xiangyu Zhu, Zheyu Yan, Xiao-Yu Zhang, Jinlin Wu, Zhen Lei, Suvidha Tripathi, Mahak Kothari, Md Haider Zama, Debayan Deb, Bernardo Biesseck, Pedro Vidal, Roger Granada, Guilherme Fickel, Gustavo Führ, David Menotti, Alexander Unnervik, Anjith George, Christophe Ecabert, Hatef Otroschi Shahreza, Parsa Rahimi, Sébastien Marcel, Ioannis Sarridis, Christos Koutlis, Georgia Baltso, Symeon Papadopoulos, Christos Diou, Nicolò Di Domenico, Guido Borghi, Lorenzo Pellegrini, Enrique Mascandela, Ángela Sánchez-Pérez, Andrea Atzori, Fadi Boutros, Naser Damer, Gianni Fenu, and Mirko Marras. Frccsyn-ongoing: Benchmarking and comprehensive evaluation of real and synthetic data to improve face recognition systems. *Information Fusion*, 107:102322, 2024. doi: 10.1016/j.inffus.2024.102322.
- [97] Venkata Sai Rishita Middi, Middi Raju, and Tanvir Ahmed Harris. Machine translation using natural language processing. *MATEC Web of Conferences*, 277:02004, 01 2019. doi: 10.1051/mateconf/201927702004.
- [98] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. doi: 10.48550/arXiv.1301.3781.
- [99] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. https://proceedings.neurips.cc/paper_files/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf.
- [100] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey, 2024. doi: 10.48550/arXiv.2402.06196.
- [101] Mihai Nadas, Laura Diosan, and Andreea Tomescu. Synthetic data generation using large language models: Advances in text and code, 2025. doi: 10.48550/arXiv.2503.14023.
- [102] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, 09 2011. doi: 10.1136/amiajnl-2011-000464.
- [103] Miguel Nunes, João Boné, João C. Ferreira, Pedro Chaves, and Luis B. Elvas. Medialbertina: An european portuguese medical language model. *Computers in Biology and Medicine*, 182:109233, 2024. doi: 10.1016/j.combiomed.2024.109233.
- [104] Lucas Emanuel Silva e. Oliveira, Ana Carolina Peters, Adalniza Moura Pucca da Silva, Caroline Pilatti Gebelua, Yohan Bonescki Gumiel, Lilian Mie Mukai Cintho, Deborah Ribeiro Carvalho, Sadid Al Hasan, and Claudia Maria Cabral Moro. Semclinbr - a multi-institutional and multi-specialty semantically annotated corpus for portuguese clinical nlp tasks. *Journal of Biomedical Semantics*, 13(1):13, May 2022. doi: 10.1186/s13326-022-00269-1.

- [105] OpenAI. Gpt-4 technical report, 2024. doi: 10.48550/arXiv.2303.08774.
- [106] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc., 2022. https://proceedings.neurips.cc/paper_files/paper/2022/file/blefde53be364a73914f58805a001731-Paper-Conference.pdf.
- [107] Pedro Henrique Paiola, Gabriel Lino Garcia, João Renato Ribeiro Manesco, Mateus Roder, Douglas Rodrigues, and João Paulo Papa. Adapting llms for the medical domain in portuguese: A study on fine-tuning and model evaluation, 2024.
- [108] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.
- [109] Cheng Peng, Xi Yang, Aokun Chen, Kaleb E. Smith, Nima PourNejatian, Anthony B. Costa, Cheryl Martin, Mona G. Flores, Ying Zhang, Tanja Magoc, Gloria Lipori, Duane A. Mitchell, Naykky S. Ospina, Mustafa M. Ahmed, William R. Hogan, Elizabeth A. Shankman, Yi Guo, Jiang Bian, and Yonghui Wu. A study of generative large language model for medical research and healthcare. *npj Digital Medicine*, 6(1):210, Nov 2023. doi: 10.1038/s41746-023-00958-w.
- [110] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. <https://aclanthology.org/D14-1162.pdf>.
- [111] Ramon Pires, Hugo Abonizio, Thales Sales Almeida, and Rodrigo Nogueira. Sabiá: Portuguese large language models. In Murilo C. Naldi and Reinaldo A. C. Bianchi, editors, *Intelligent Systems*, pages 226–240, Cham, 2023. Springer Nature Switzerland.
- [112] Ekaterina Plesovskaya and Sergey Ivanov. An empirical analysis of kde-based generative models on small datasets. *Procedia Computer Science*, 193:442–452, 2021. 10th International Young Scientists Conference in Computational Science, YSC2021, 28 June – 2 July, 2021.
- [113] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [114] Lance A. Ramshaw and Mitchell P. Marcus. Text chunking using transformation-based learning, 1995.

- [115] Shaina Raza and Brian Schwartz. Entity and relation extraction from clinical case reports of covid-19: a natural language processing approach. *BMC Medical Informatics and Decision Making*, 23(1):20, Jan 2023. doi: 10.1186/s12911-023-02117-3.
- [116] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. <https://fq.pkwyx.com/default/https/aclanthology.org/D19-1410.pdf>, 2019.
- [117] João Rodrigues, Luís Gomes, João Silva, António Branco, Rodrigo Santos, Henrique Lopes Cardoso, and Tomás Osório. Advancing neural encoding of portuguese with transformer albertina pt-*. In Nuno Moniz, Zita Vale, José Cascalho, Catarina Silva, and Raquel Sebastião, editors, *Progress in Artificial Intelligence*, pages 441–453, Cham, 2023. Springer Nature Switzerland. doi: 10.1007/978-3-031-49008-8_35.
- [118] Sunita Sarawagi. Information extraction. *Foundations and Trends® in Databases*, 1(3):261–377, 2008. doi: 10.1561/19000000003.
- [119] Atiquer Rahman Sarkar, Yao-Shun Chuang, Noman Mohammed, and Xiaoqian Jiang. De-identification is not enough: a comparison between de-identified and synthetic clinical notes. *Scientific Reports*, 14(1):29669, Nov 2024. doi: 10.1038/s41598-024-81170-y.
- [120] Elisa Terumi Rubel Schneider, João Vitor Andrioli de Souza, Julien Knafo, Lucas Emanuel Silva e Oliveira, Jenny Copara, Yohan Bonescki Gumiel, Lucas Ferro Antunes de Oliveira, Emerson Cabrera Paraiso, Douglas Teodoro, and Cláudia Maria Cabral Moro Barra. BioBERTpt - a Portuguese neural language model for clinical named entity recognition. In Anna Rumshisky, Kirk Roberts, Steven Bethard, and Tristan Naumann, editors, *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 65–72, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.clinicalnlp-1.7.
- [121] Elisa Terumi Rubel Schneider, João Vitor Andrioli de Souza, Yohan Bonescki Gumiel, Claudia Moro, and Emerson Cabrera Paraiso. A gpt-2 language model for biomedical texts in portuguese. In *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)*, pages 474–479, 2021. doi: 10.1109/CBMS52027.2021.00056.
- [122] Elliot Schumacher, Andriy Mulyar, and Mark Dredze. Clinical concept linking with contextualized neural representations. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8585–8592, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.760.
- [123] Veronica Sciannameo, Daniele Jahier Pagliari, Sara Urru, Piercesare Grimaldi, Honoria Ocagli, Sara Ahsani-Nasab, Rosanna Irene Comoretto, Dario Gregori, and Paola Berchiolla. Information extraction from medical case reports using openai instructgpt. *Computer Methods and Programs in Biomedicine*, 255:108326, 2024. doi: 10.1016/j.cmpb.2024.108326.
- [124] Jingpu Shi, Dong Wang, Gino Tesei, and Beau Norgeot. Generating high-fidelity privacy-conscious synthetic patient data for causal effect estimation with multiple treatments. *Frontiers in Artificial Intelligence*, Volume 5 - 2022, 2022. doi: 10.3389/frai.2022.918813.
- [125] Jiyun Shi, Zhimeng Yuan, Wenxuan Guo, Chen Ma, Jiehao Chen, and Meihui Zhang. Knowledge-graph-enabled biomedical entity linking: a survey. *World Wide Web*, 26(5):2593–2622, Sep 2023. doi: 10.1007/s11280-023-01144-4.

- [126] Ilya Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, Jul 2024. doi: 10.1038/s41586-024-07566-y.
- [127] Julius Sim and Chris C Wright. The kappa statistic in reliability studies: Use, interpretation, and sample size requirements. *Physical Therapy*, 85(3):257–268, 03 2005. doi: 10.1093/ptj/85.3.257.
- [128] Sonit Singh. Natural language processing for information extraction, 2018. doi: 10.48550/arXiv.1807.02383.
- [129] Ayush Singhal, Michael Simmons, and Zhiyong Lu. Text mining genotype-phenotype relationships from biomedical literature for database curation and precision medicine. *PLOS Computational Biology*, 12(11):1–19, 11 2016. doi: 10.1371/journal.pcbi.1005017.
- [130] Diana Sousa. Deep learning system for biomedical relation extraction combining external sources of knowledge. In Djoerd Hiemstra, Marie-Francine Moens, Josiane Mothe, Raffaele Perego, Martin Potthast, and Fabrizio Sebastiani, editors, *Advances in Information Retrieval*, pages 688–693, Cham, 2021. Springer International Publishing. doi: 10.1007/978-3-030-72240-1_82.
- [131] Hugo Sousa, Alipio Mario Jorge, Arian Pasquali, Catarina Santos, and Mario Lopes. A biomedical entity extraction pipeline for oncology health records in portuguese. In *Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing*, SAC ’23, page 950–956, New York, NY, USA, 2023. Association for Computing Machinery. doi: 10.1145/3555776.3578577.
- [132] Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. Bertimbau: Pretrained bert models for brazilian portuguese. In Ricardo Cerri and Ronaldo C. Prati, editors, *Intelligent Systems*, pages 403–417, Cham, 2020. Springer International Publishing. doi: 10.1007/978-3-030-61377-8_28.
- [133] Supriyono, Aji Prasetya Wibawa, Suyono, and Fachrul Kurniawan. A survey of text summarization: Techniques, evaluation and challenges. *Natural Language Processing Journal*, 7:100070, 2024. doi: 10.1016/j.nlp.2024.100070.
- [134] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. https://proceedings.neurips.cc/paper_files/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf.
- [135] Sakirin Tam, Rachid Ben Said, and Ö. Özgür Tanriöver. A convbilstm deep learning model-based approach for twitter sentiment classification. *IEEE Access*, 9:41283–41293, 2021. doi: 10.1109/ACCESS.2021.3064830.
- [136] Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. Does synthetic data generation of llms help clinical text mining?, 2023. doi: 10.48550/arXiv.2303.04360.
- [137] Wilson L. Taylor. “cloze procedure”: A new tool for measuring readability. *Journalism Quarterly*, 30(4):415–433, 1953. doi: 10.1177/107769905303000401.

- [138] Gemini Team. Gemini: A family of highly capable multimodal models, 2024. doi: 10.48550/arXiv.2312.11805.
- [139] Gemma Team. Gemma: Open models based on gemini research and technology, 2024. doi: 10.48550/arXiv.2403.08295.
- [140] Llama Team. The llama 3 herd of models, 2024. doi: 10.48550/arXiv.2407.21783.
- [141] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023. doi: 10.48550/arXiv.2302.13971.
- [142] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023. doi: 10.48550/arXiv.2307.09288.
- [143] Elena Tutubalina, Zulfat Miftahutdinov, Sergey Nikolenko, and Valentin Malykh. Medical concept normalization in social media posts with recurrent neural networks. *Journal of Biomedical Informatics*, 84:93–102, 2018. doi: 10.1016/j.jbi.2018.06.006.
- [144] Fahad Umer and Niha Adnan. Generative artificial intelligence: synthetic datasets in dentistry. *BDJ Open*, 10(1):13, Mar 2024. doi: 10.1038/s41405-024-00198-4.
- [145] Boris van Breugel, Tennison Liu, Dino Oglic, and Mihaela van der Schaar. Synthetic data in biomedicine via generative artificial intelligence. *Nature Reviews Bioengineering*, 2(12):991–1004, Dec 2024. doi:10.1038/s44222-024-00245-7.
- [146] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. <https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf?fbclid=...>, 2008.
- [147] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- [148] Anthony J Viera, Joanne M Garrett, et al. Understanding interobserver agreement: the kappa statistic. *Fam med*, 37(5):360–363, 2005. http://www1.cs.columbia.edu/~julia/courses/CS6998/Interrater_agreement.Kappa_statistic.pdf.

- [149] Alina Vretinaris, Chuan Lei, Vasilis Efthymiou, Xiao Qin, and Fatma Özcan. Medical entity disambiguation using graph neural networks. In *Proceedings of the 2021 International Conference on Management of Data, SIGMOD '21*, page 2310–2318, New York, NY, USA, 2021. Association for Computing Machinery. doi: 10.1145/3448016.3457328.
- [150] Yanshan Wang, Liwei Wang, Majid Rastegar-Mojarad, Sungrim Moon, Feichen Shen, Naveed Afzal, Sijia Liu, Yuqun Zeng, Saeed Mehrabi, Sunghwan Sohn, and Hongfang Liu. Clinical information extraction applications: A literature review. *Journal of Biomedical Informatics*, 77:34–49, 2018. doi: 10.1016/j.jbi.2017.11.011.
- [151] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions, 2023. doi: 10.48550/arXiv.2212.10560.
- [152] Qiang Wei, Zongcheng Ji, Yuqi Si, Jingcheng Du, Jingqi Wang, Firat Tiryaki, Stephen Wu, Cui Tao, Kirk Roberts, and Hua Xu. Relation extraction from clinical narratives using pre-trained language models. *AMIA Annu Symp Proc*, 2019:1236–1245, March 2020. <https://pmc.ncbi.nlm.nih.gov/articles/PMC7153059/>.
- [153] John Weldon, Tomas Ward, and Eoin Brophy. Generation of synthetic electronic health records using a federated gan, 2021. doi: 10.48550/arXiv.2109.02543.
- [154] Maciej Wiatrak and Juha Iso-Sipila. Simple hierarchical multi-task neural end-to-end entity linking for biomedical text. In Eben Holderness, Antonio Jimeno Yepes, Alberto Lavelli, Anne-Lyse Minard, James Pustejovsky, and Fabio Rinaldi, editors, *Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis*, pages 12–17, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.louhi-1.2.
- [155] Daya C. Wimalasuriya and Dejing Dou. Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science*, 36(3):306–323, 2010. doi: 10.1177/0165551509360123.
- [156] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google’s neural machine translation system: Bridging the gap between human and machine translation, 2016. doi: 10.48550/arXiv.1609.08144.
- [157] Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. Large language models for generative information extraction: a survey. *Frontiers of Computer Science*, 18(6):186357, Nov 2024. doi: 10.1007/s11704-024-40555-y.
- [158] Ran Xu, Hejie Cui, Yue Yu, Xuan Kan, Wenqi Shi, Yuchen Zhuang, May Dongmei Wang, Wei Jin, Joyce Ho, and Carl Yang. Knowledge-infused prompting: Assessing and advancing clinical text data generation with large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics ACL 2024*, pages 15496–15523, Bangkok, Thailand and virtual meeting, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.916.

- [159] Liu Xue, Song Qing, and Zhang Pengzhou. Relation extraction based on deep learning. In *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*, pages 687–691, 2018. doi: 10.1109/ICIS.2018.8466437.
- [160] Yuchong Yao, Xiaohui Wang, Yuanbang Ma, Han Fang, Jiaying Wei, Liyuan Chen, Ali Anaissi, and Ali Braytee. Conditional variational autoencoder with balanced pre-training for generative adversarial networks. In *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10, 2022. doi: 10.1109/D-SAA54385.2022.10032367.
- [161] Janine Young, Paul Graham, and Rebecca Penny. Using bayesian networks to create synthetic data. *Journal of Official Statistics*, 25(1):1–17, 2009. <https://www.scb.se/contentassets/ca21efb41fee47d293bbee5bf7be7fb3/using-bayesian-networks-to-create-synthetic-data.pdf>.
- [162] Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander J Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. Large language model as attributed training data generator: A tale of diversity and bias. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 55734–55784. Curran Associates, Inc., 2023.
- [163] Klim Zaporozhets, Johannes Deleu, Chris Develder, and Thomas Demeester. Dwie: An entity-centric dataset for multi-task document-level information extraction. *Information Processing Management*, 58(4):102563, 2021. doi: 10.1016/j.ipm.2021.102563.
- [164] Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. Central moment discrepancy (cmd) for domain-invariant representation learning, 2019. doi: 10.48550/arXiv.1702.08811.
- [165] Gongbo Zhang, Qiao Jin, Yiliang Zhou, Song Wang, Betina Idnay, Yiming Luo, Elizabeth Park, Jordan G. Nestor, Matthew E. Spotnitz, Ali Soroush, Thomas R. Campion, Zhiyong Lu, Chunhua Weng, and Yifan Peng. Closing the gap between open source and commercial large language models for medical evidence summarization. *npj Digital Medicine*, 7(1):239, Sep 2024. doi: 10.1038/s41746-024-01239-w.
- [166] Sheng Zhang, Hao Cheng, Shikhar Vashishth, Cliff Wong, Jinfeng Xiao, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Knowledge-rich self-supervised entity linking. *CoRR*, abs/2112.07887, 2021. doi: 10.48550/arXiv.2112.07887.
- [167] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023. doi: 10.48550/arXiv.2303.18223.
- [168] Jun Zhu. Synthetic data generation by diffusion models. *National Science Review*, 11(8):nwae276, 08 2024. doi: 10.1093/nsr/nwae276.
- [169] Ming Zhu, Busra Celikkaya, Parminder Bhatia, and Chandan K. Reddy. Latte: Latent type modeling for biomedical entity linking. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9757–9764, Apr. 2020. doi: 10.1609/aaai.v34i05.6526.

Appendix A

Engineered prompts

In this appendix, we display the prompts that the studies described in [Section 3.2.2](#) used to generate their respective datasets. These prompts are the final result of the prompt engineering process of those studies.

Write a discharge summary with a detailed hospital course for a patient with these conditions: (list of conditions)

The patient underwent these procedures: (list of procedures)

The Discharge Summary should have a word limit of 4000 words. The names of people and locations within the discharge summary should be de-identified. Do not state ICD-10 codes in the main body of the text. For any condition involving a numeric range, state explicitly a number within that range (e.g., for a patient with blood glucose between 7.0 and 11.0 mmol/l generate “patient’s glucose level was 8.6 mmol/l”). For conditions with the keyword “other” specify the condition that falls into the set of “other” - e.g., for “Other specified congenital malformations of skin” generate “Aplasia cutis congenita”. For conditions with the keyword “unspecified”, do not include the word “unspecified” within the main body of text - e.g., for the code H35.00 (Unspecified background retinopathy) generate the statement “the patient suffers from background retinopathy”. At the end of the discharge summary add a paragraph with the header “DISCHARGE DIAGNOSES AND PROCEDURES” and assign ICD-10 codes to the discharge diagnoses and procedures, for each concept stating its ICD-10 code and its description (e.g., Essential (Primary) Hypertension [I10]).

Listing A.1: Template of prompt used by Falis et al. [\[30\]](#). Source: Falis et al. [\[30\]](#).

Please act as a sentence generator for the biological domain and provide 30 sentences containing the words [Seed Entities]. These sentences should not include any additional information or explanation. Generated sentences should mimic the style of PubMed journal articles, using a variety of sentence structures.

Listing A.2: Prompt used by Tang et al. [136]. Source: Tang et al. [136].

Create a narrative psychiatric clinical note (about 500 words) and annotate sentences with PATIENT-RELATED factors (older age, family member mental disorder, abuse, childhood abuse), ILLNESS-RELATED factors (long illness duration, severe illness, suicidality, multiple hospitalizations, recurrent episodes, improvement, physical comorbidity, mental comorbidity, substance abuse, anhedonia, illness early onset) and TREATMENT-RELATED factors (multiple antidepressants, antidepressant dosage increase, multiple psychotherapies, side effects, non adherence). Make sure to annotate all relevant text for each factor. The annotation must indicate if the evidence is POSITIVE (presence of factor) or NEGATIVE (absence of factor) in the exact same format as the following examples: "The patient is a 32-year old male [PATIENT_FACTOR(NEGATIVE):older_age]." "The patient reports a history of mental illness in her family, with her brother suffering from generalised anxiety [PATIENT_FACTOR(POSITIVE):family_member_mental_disorder] and that she visits him regularly."; "She experienced childhood abuse, suffering emotional neglect from her mother for several years [PATIENT_FACTOR(POSITIVE):childhood_abuse]."; Annotate each factor separately within each sentence after the relevant text, e.g.: "The dosage was gradually increased [TREATMENT_FACTOR(POSITIVE):antidepressant_dosage_increase] until severe gastrointestinal distress emerged [TREATMENT_FACTOR(POSITIVE):side_effects] leading to a review of current approach." "She experienced multiple periods of depression [ILLNESS_FACTOR(POSITIVE):recurrent_episodes] but was never hospitalized [ILLNESS_FACTOR(NEGATIVE):multiple_hospitalizations] as she opposed it repeatedly."; "He was treated with various antidepressants [ILLNESS_FACTOR(POSITIVE):multiple_antidepressants] and experienced relapses upon discontinuation [ILLNESS_FACTOR(POSITIVE):non_adherence] which was due to a general distrust in medications." Do not annotate the plan.

Listing A.3: Prompt used by Lorge et al. [89]. Source: Lorge et al. [89].

Suppose you need to create a dataset for disease recognition. Your task is to: 1. Generate a sentence about disease. 2. Output a list of named entity about disease only. 3. The sentence should mention the disease named Specific Disease. 4. The sentence should mimic the style of clinical reports.

Some examples are: Sentence: [Sentence related to Specific Disease] Disease: [Named entities related to Specific Disease found in sentence]

Listing A.4: Template of prompt used by Xu et al. [158]. Source: Xu et al. [158].