Synthetic4Health: generating annotated synthetic clinical letters

Ren, L.; Belkadi, S.; Han, L.; Del-Pinto, W.; Nenadic, G.

frontiers

SYNTHETIC4HEALTH: Generating Annotated Synthetic Clinical Letters

Libo Ren [1], **Samuel Belkadi** [2], **Lifeng Han** [1,3*], **Warren Del-Pinto** [1], **and Goran Nenadic** [1]

[1] *University of Manchester, Oxford Rd, Greater Manchester, UK*
[2] *Department of Engineering, University of Cambridge, UK*
[3] *LIACS & LUMC, Leiden University, NL*

Correspondence*:
Corresponding Author
l.han@lumc.nl — lifeng.han@manchester.ac.uk

ABSTRACT

Since clinical letters contain sensitive information, clinical-related datasets cannot be widely applied in model training, medical research, and education. This work aims to generate reliable, diverse, and de-identified synthetic clinical letters. We explored various pre-trained language models (PLMs) for text masking and generation. After that, we worked on Bio_ClinicalBERT, a high-performing model, and experimented with different masking strategies. Both qualitative and quantitative methods were used for evaluation. A downstream task, Named Entity Recognition (NER), was further implemented to assess the usability of these synthetic letters. We also applied clinically focused evaluation methods - including BioGPT and GPT-3.5-turbo - to evaluate the clinical performance of the synthetic texts.

The results indicate that 1) encoder-only models outperform encoder-decoder models. 2) Among encoder-only models, those trained on general corpora perform comparably to those trained on clinical data when clinical information is preserved. 3) Preserving clinical entities and document structure aligns more closely with our objectives than simply fine-tuning the model. 4) Masking strategies have a noticeable impact on the quality of synthetic clinical letters: masking stopwords has a positive impact, while masking nouns or verbs has a negative effect. 5) BERTScore should be the primary quantitative evaluation metric, with other metrics serving as supplementary references. 6) Contextual information has only a limited effect on the models' understanding, suggesting that synthetic letters can effectively substitute real ones in downstream NER tasks. 7) Although the model occasionally generates hallucinated content, it appears to have little effect on overall clinical performance.

Unlike previous research, which primarily focuses on reconstructing the original letters by training language models, this paper provides a foundational framework for generating diverse, de-identified clinical letters. It offers a direction for utilising the model to process real-world clinical letters, thereby helping to expand datasets in the clinical domain. Our codes and trained models are available at `https://github.com/HECTA-UoM/Synthetic4Health`

Keywords: Pre-trained Language Models (PLMs), Encoder-Only Models, Encoder-Decoder Models, Masking and Generating; Named Entity Recognition (NER)

1 INTRODUCTION

With the development of medical information systems, electronic clinical letters are increasingly used in communication between healthcare departments. These clinical letters typically contain detailed information about patients' visits, including their symptoms, medical history, medications, etc (Rayner et al., 2020). They also often include sensitive information, such as patients' names, phone numbers, and addresses (Tarur and Prasanna, 2021; Tucker et al., 2016). As a result, these letters are difficult to share and nearly impossible to use widely in clinical education and research.

In 2018, 325 severe breaches of protected health information were reported by CynergisTek (Abouelmehdi et al., 2018). Among these, nearly 3,620,000 patients' records were at risk (Abouelmehdi et al., 2018). This is just the data from one year—similar privacy breaches are unfortunately common. The most severe hacking incident affected up to 16,612,985 patients (Abouelmehdi et al., 2018). Therefore, generating synthetic letters and applying de-identification techniques seem to be indispensable.

Additionally, due to privacy concerns and access controls, insufficient data is the major challenge of clinical education, medical research, and system development (Spasic and Nenadic, 2020). Some shared datasets offer de-identified annotated data. The MIMIC series is a typical example. These datasets are accessible through PhysioNet. MIMIC-IV (Johnson et al., 2023b; A et al., 2000; Johnson et al., 2024a), the latest version, contains data from 364,627 patients' clinical information collected from 2008 to 2019 at a medical centre in Boston. It records details about hospitalisations, demographics, and transfers. Numerous research are based on this shared dataset. Another public dataset series in the clinical domain is i2b2/n2c2 (the President and of Harvard College, 2023), They are accessible through the DBMI Data Portal. This series includes unstructured clinical notes such as process notes, radiology reports, and discharge summaries, and is published for clinical informatics sharing and NLP (Natural Language Processing) tasks challenges.

However, these sharing datasets are often limited to specific regions and institutions, making them not comprehensive. Consequently, models and medical research outcomes derived from these datasets cannot be widely applied (Humbert-Droz et al., 2022). Therefore, to address the lack of clinical datasets and reduce the workload for clinicians, it is essential to explore available technologies that can automatically generate de-identified clinical letters.

Existing systems generate clinical letters primarily by integrating structured data, while there are not many studies on how to use Natural Language Generation (**NLG**) models for this task (HÜSKE-KRAUS, 2003; Amin-Nejad et al., 2020a; Tang et al., 2023). NLG attempts to combine clinical knowledge with general linguistic expressions and aims to generate clinical letters that are both readable and medically sound. However, NLG technology is not yet mature enough for widespread use in healthcare systems. Additionally, it faces numerous challenges, including medical accuracy, format normalisation, and de-identification (HÜSKE-KRAUS, 2003). Therefore, this investigation focuses on how NLG technology can be used to generate reliable and anonymous clinical letters, which can benefit medical research, clinical education, and clinical decision-making.

The main aim of our work is to *generate de-identified clinical letters* that can *preserve clinical information* while *differing from the original* letters. A brief example of our objective is shown in Figure 1. Based on this objective, different generation models will be explored as a preliminary attempt. Then we select the best models and try various techniques to improve the quality of the synthetic letters. The synthetic letters are evaluated not only with quantitative and qualitative methods, but also in downstream tasks, i.e., Named

Entity Recognition (NER). We hope this work will contribute to addressing the challenge of insufficient data in the clinical domain.

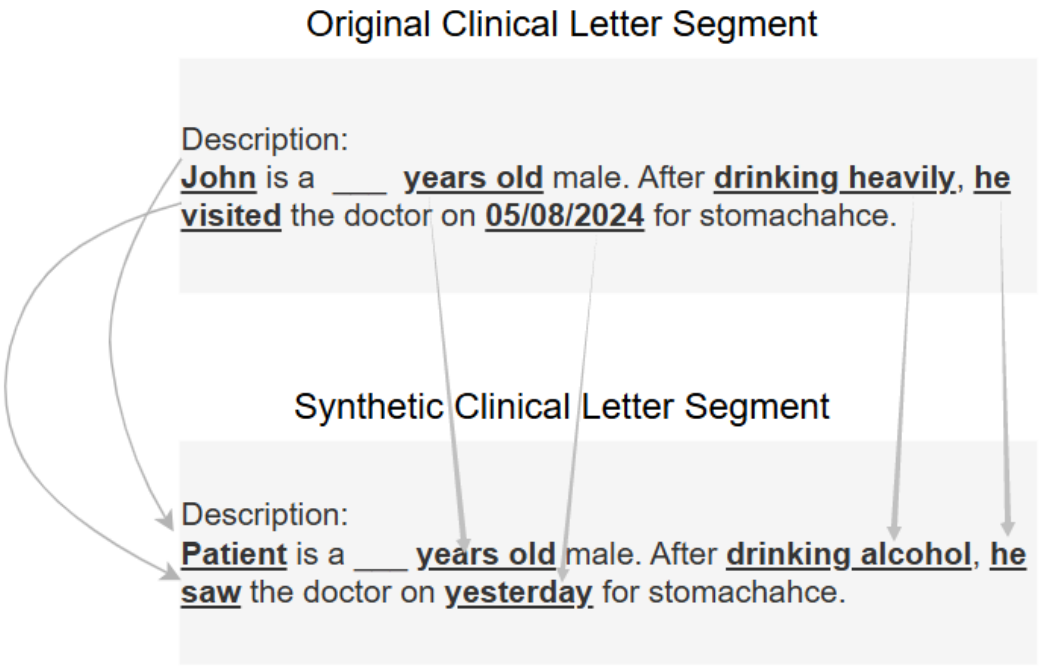

Figure 1. An Example of the Objective: sentence/segment-level generations

In summary, this work is centered on the Research Question (RQ): "How can we generate reliable and diverse clinical letters without including sensitive information?" Specifically, it will answer the following related sub-questions (RQs):

1. How do different models perform in masking and generating clinical letters?
2. How should the text be segmented in clinical letter generation?
3. How do different masking strategies affect the quality of synthetic letters?
4. How can we evaluate the quality of synthetic letters?

To answer these questions, we explored various LLMs (Large Language Models) for masking and generating clinical letters, ultimately focusing on one that performed well. The overall **highlights** of this work are summarised as follows:

1. Mask and Generate clinical letters using different LLMs at the sentence level.
2. Explore methods to improve synthetic clinical letters' readability and clinical soundness.
3. Initially evaluate synthetic letters using both qualitative and quantitative methods.
4. Apply synthetic letters in downstream tasks and further evaluate them using clinically focused methods.
5. Explore post-processing methods to further enhance the quality of de-identified letters.

2 BACKGROUND AND LITERATURE REVIEW

We first introduce general language models, followed by their applications, especially within the clinical domain. We then present the generative language models based on the Transformer architecture. These models serve as the technical foundation for most modern text generation tasks. Afterward, We review related works, discussing their relevance and how they are connected to ours. Finally, all quantitative evaluation metrics used in this paper are introduced.

2.1 Development of Language Models (LMs)

The development of Language Models can be divided into three stages: Rule-Based Approach, Supervised Modelling, and Unsupervised Modelling (Boonstra et al., 2024).

2.1.1 Rule-Based Approach

The Rule-Based Approach was first used in 1950s, which marks the beginning of NLP (Nadkarni et al., 2011). This approach always uses a set of predefined rules, which were written and maintained manually by specialists (van der Lee et al., 2018; Satapathy et al., 2017). Although it can generate standardised text without being fed with extensive input data (van der Lee et al., 2018), there are still numerous limitations. Initially, manually crafted rules are often ambiguous, and the dependencies between different rules increase maintenance costs (Nadkarni et al., 2011). Secondly, these stylised models cannot perform well in understanding realistic oral English and ungrammatical text such as clinical discharge records, although these texts are still readable for humans (Nadkarni et al., 2011). Thirdly, they are not objective enough, as they are affected by the editors of the rule library. Additionally, they are not flexible enough to deal with special cases. Therefore, the Rule-Based Method is only suitable for analysing and generating highly standardised texts like prescriptions (van der Lee et al., 2018).

2.1.2 Supervised Language Models

To address the limitations of the Rule-Based Approach, Supervised Learning has been applied to NLP. The invention of Statistical Machine Translation (SMT) in 1990 marked the rise of supervised NLP (Boonstra et al., 2024). It learns the correspondence rules between different languages by analysing the input of bilingual texts (parallel corpus) (Brown et al., 1990). Supervised NLP models are trained on annotated labels to learn rules automatically. The learned rules will be used in word prediction or text classification. Hidden Markov Model (HMM) and Conditional Random Field (CRF) are two typical applications of this stage (Sharma et al., 2023). Both of them worked by tagging features of the input texts. HMM generates data by statistically analysing word frequencies (Eddy, 1996; Masuko et al., 1998). CRF, however, searches globally and calculates joint probabilities to get an optimal solution (Sutton et al., 2012; Bundschus et al., 2008). Long Short-Term Memory (LSTM) is another typical example of supervised language modeling (Hochreiter and Schmidhuber, 1997). In the task of text generation, the input should be a set of labelled data or word vector sequences. By minimising the loss between the predicted word vector and the actual word vector, LSTM can capture the dependencies between words in long texts (Santhanam, 2020; Graves and Graves, 2012).

Although Supervised Language Models performs better than the Rule-Based Approach, domain experts still need to annotate the training dataset (Boonstra et al., 2024). In addition, data in some domains are difficult to collect due to privacy issues (such as **medical** and *legal* domains). This became an ongoing challenge in applying the supervised language models to specific tasks.

2.1.3 Unsupervised Language Models

To address the high cost and difficulty of obtaining labelled data, unsupervised Neural Networks are applied to the language modelling (Bengio et al., 2000). The popularity of corpora such as Wikipedia and social media provides enough data for unsupervised models' training (Boonstra et al., 2024). Word embedding is a significant technique in this stage (Johnson et al., 2024b). **Word2Vec** represents words using vectors with hundreds of dimensions. The context can be captured by training word vectors in the sliding window. By adjusting the hyperparameters to maximise the conditional probability of the target word, it can learn semantic information accurately (Mikolov et al., 2013a,b) (e.g. 'Beijing'-'China'+'America' => 'Washington' (Ma and Zhang, 2015)). After training, each word usually has a fixed word vector regardless of the context in which it appears (known as Static Word Embedding) (Santhanam, 2020).

Unlike word2Vec, **BERT** and **GPT** use contextual word embedding. Their word vectors reflect the semantic information and are affected by the context (Camacho-Collados and Pilehvar, 2018). BERT focuses on contextual understanding (Zhang et al., 2020) (e.g. The bank is full of lush willows, where 'bank' means the riverside, not the financial institution), while GPTs focus on text generation in a specific context (Liu et al., 2022; Li et al., 2024) (e.g. Prompt: "Do you know Big Ben?" Answer: "Yes, I know Big Ben. It is the nickname for the Great Bell of the Clock located in London.") Although unsupervised language models have been able to train and understand text proficiently, they still face challenges in practical applications, such as difficulty handling ambiguity and high computing resource consumption. Therefore, language modelling still has a long way to go.

2.2 Language Models Applications in Clinical Domain

Based on the modelling methods mentioned above, a variety of language models have been invented. They play an important role in scientific research and daily life, especially in the field of **healthcare**. In this section, we will discuss the *clinical language model* applications in detail from two aspects: named entity recognition (**NER**) and natural language generation (**NLG**).

2.2.1 Named Entity Recognition (NER)

NER was originally designed for text analysis and recognition of named entities, such as dates, organisations, and proper nouns (Grishman and Sundheim, 1996). In the clinical domain, NER is used to identify *clinical events* (e.g. symptoms, drugs, treatment plans, etc.) from unstructured documents with their *qualifiers* (e.g. chronic, acute, mild), classify them, and extract the relationship (Bose et al., 2021; Kundeti et al., 2016). Initially, NER relied on rule-based and machine-learning methods that required extensive manual feature engineering. In 2011, (Collobert et al., 2011) used word embeddings and neural networks in NER. Since then, research in NER has shifted to automatic feature extraction.

SpaCy [1] is an open-source NLP library for tasks like POS tagging and text classification. Additionally, it offers a range of pre-trained NER models. **ScispaCy** [2], a fine-tuned extension of spaCy on medical science datasets, can recognise entities such as "DISEASE", "CHEMICAL", and "CELL", which are essential for medical research. Although NER is useful in rapidly extracting clinical terms, there are still a lot of challenges, such as **non-standardisation** (extensive use of abbreviated words in clinical texts), **misspellings** (due to manual input by medical staff), and **ambiguity** (often influenced by context, e.g.whether 'back' refers to an adverb or an anatomical entity) (Bose et al., 2021). Existing research

[1] https://spacy.io/

[2] https://allenai.github.io/scispacy/

mitigates these problems using *entity linking* (mapping extracted clinical entities to medical repositories such as UMLS and SNOMED). More deep-learning models and text analysis tools are being developed to solve these issues.

2.2.2 De-Identification

The unprocessed clinical text has a risk of personal information leakage. Additionally, manual de-identification is not only prone to errors but also costly. Therefore, research on de-identification is indispensable for the secondary use of clinical data. Typically, de-identification is based on NER models to identify **Protected Health Information (PHI)**. Then PHI will be processed by different strategies (such as synonym replacement, removal, or masking) (Berg et al., 2021, 2020).

Similar to NER, early de-identification relied heavily on rule-based systems, machine learning, or hybrid models. Physionet DeID, the VHA Best-of-Breed (BoB), and MITRE's MIST are three typical examples (Meystre, 2015). However, these algorithms require extensive handcrafted feature engineering. With the development of unsupervised learning, recurrent neural networks (RNNs) and Transformers are widely used in de-identification tasks (Dernoncourt et al., 2017; Kovačević et al., 2024).

Philter, a Protected Health Information filter (Norgeot et al., 2020), is a pioneering system that combines rule-based approaches with state-of-the-art NLP models to identify and remove PHI. Although Philter outperforms many existing tools like Physionet and Scrubber, particularly in recall and F2 score, it still requires large amounts of annotated data for training (Norgeot et al., 2020). Additionally, research has shown that while the impact of de-identification on downstream tasks is minimal, it cannot be completely ignored (Meystre et al., 2014). Therefore, performing de-identification without mistakenly removing semantic information is still a challenge in this field.

2.2.3 Natural Language Generation (NLG)

Both label-to-text and text-to-text generation are components of NLG (Gatt and Krahmer, 2018). NLG consists of six primary sub-tasks, covering most of the NLG process. NLG architectures can generally be divided into three categories (Gatt and Krahmer, 2018):

- **Modular Architectures:** This architecture consists of three modules: the Text Planner (responsible for determining the content for generation), the Sentence Planner (which aggregates the synthetic text), and the Realiser (which generates grammatically correct sentences). These modules are closely related to the six sub-tasks, and each module operates independently.
- **Planning Perspectives:** This architecture considers NLG as a planning problem. It generates tokens dynamically based on the objectives, with potential dependencies between different steps.
- **Integrated or Global Approaches:** This is the dominant architecture for NLG, relying on statistical learning and deep learning. Common generative models, such as Transformers and conditional language models, are included in this architecture.

In the field of healthcare, NLG applications include document generation and question-answering. Document generation involves discharge letters, diagnostic reports for patients, decision-making suggestions for experts, and personalised patient profiles for administrators (Cawsey et al., 1997). Some systems have already been implemented in practice. For instance, PIGLIT generates explanations of clinical terminology for diabetes patients (Hirst et al., 1997), and MAGIC can generate reports for Intensive Care Unit (ICU) patients (McKeown et al., 1997). Question answering is another application of NLG. Tools like chatbots can provide patients with answers to basic healthcare questions (Locke et al., 2021).

Nowadays, NLG in the clinical field focuses on the development and training of transformer-based large language models (LLMs); examples of this work can be seen in (Amin-Nejad et al., 2020a; Luo et al., 2022). These models perform well in specific domains such as semantic query (Kong et al., 2022) and electronic health records (**EHRs**) generation (Lee, 2018). However, very few systems can reliably produce concise, readable, and clinically sound reports across multiple sub-domains (Cawsey et al., 1997).

2.3 Generative Language Models

2.3.1 Transformer and Attention Mechanism

Although Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) are effective at semantic understanding, their recursive structure not only prevents parallel computation, but also makes them prone to gradient vanishing (Gillioz et al., 2020). The introduction of the Transformer in 2017 addressed this issue by replacing the recurrent structure with a multi-head attention mechanism (Vaswani et al., 2017). Since then, most deep learning models have been based on the Transformer. Transformer architecture is based on an encoder-decoder model (Vaswani et al., 2017). To understand this, we first need to overview Auto-Regressive Models and the Multi-Head Attention Mechanism.

Auto-Regressive Models Predictions for each Auto-Regressive Model token depend on the previous output. Therefore, it can only access the preceding tokens and operate iteratively. When the input sequence is *X*, the Auto-Regressive Model aims to train parameters θ to maximise the log-likelihood of the conditional probability P (Vaswani et al., 2017).

$$L(X) = \sum_{i} \log P(x_i \mid x_{i-k}, \ldots, x_{i-1}; \Theta) \tag{1}$$

Multi-Head Attention Mechanism Attention mechanism was initially proposed by (Cho et al., 2014). It can not only focus on the element being processed, but also capture the context dependence (Vaswani et al., 2017). Multi-head attention consists of several single-head attention (Scaled Dot-Product Attention) layers (Vaswani et al., 2017). Each word in the input sequence is converted into a high-dimensional vector representing semantic information by word embedding. These vectors pass linear transformation layers and get vectors for queries (Q), keys (K), and values (V). For each word, Q, K, and V are inputs to this single-head attention layer. The importance score of this word is calculated, and V corresponding to this word should be multiplied to get the output of this head (called Attention). Finally, outputs from all layers are concatenated to form a larger vector, which is the input to a feed-forward neural network (also the output of the multi-head attention layer) (Vaswani et al., 2017).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2}$$

Transformer and Pre-training Language Models (PLMs) Transformer consists of an encoder and a decoder. The Auto-Regressive Model is the basis of the decoder. When the input sequence is $X = (x_1, \ldots, x_N)$, output sequence is $Y_M = (y_1, \ldots, y_M)$, the model can learn a latent feature representation $Z = (z_1, \ldots, z_N)$ from X to Y. The generation of each new element Y_M relies on the generated sequence $Y_{M-1} = (y_1, \ldots, y_{M-1})$ and feature representation Z. Both the encoder and the decoder use the multi-head attention mechanism (Vaswani et al., 2017; Gillioz et al., 2020).

Many modern models are based entirely or partially on the Transformer. They compute general feature representations for the training set by unsupervised learning. This is the concept of Pre-training Language Models (PLMs). They can be fine-tuned to adapt to the specific tasks on particular datasets (Gillioz et al., 2020; Li et al., 2024).

2.3.2 Encoder-Only Models

Since the Transformer's encoder architecture can effectively capture the semantic features, some models only use this part for training. They are applied in text understanding tasks, such as text classification and NER. Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) is a representative model among them.

Unlike the Transformer decoder, which uses an Auto-Regressive model, BERT is trained based on the Masked Language Model (MLM) (Li et al., 2024). It masks the word in the input sequence, and uses the bidirectional encoder to understand the context semantically, which will be used in predicting the masked word (Devlin et al., 2019). It has already been pre-trained on a 16GB corpus. To deploy it, we only need to replace the original fully connected layer with a new output layer, and then fine-tune the parameters on the dataset for specific tasks (Devlin et al., 2019). This approach consumes fewer computing resources and less time than training a model from scratch. In the clinical domain, Bio_ClinicalBERT (Alsentzer et al., 2019) and medicalai/ClinicalBERT (Wang et al., 2023) are fine-tuned in the clinical dataset based on BERT architecture. Initially, due to the BERT's focus on semantic understanding, it was rarely used for text generation (Yang et al., 2019).

Robustly Optimized BERT Pretraining Approach (RoBERTa) (Zhuang et al., 2021) improved some key hyperparameters based on BERT. Instead of BERT's static mask, it uses a dynamic mask strategy, which helps it better adapt to multitasking. Additionally, it gained a stronger semantic understanding after training on five English datasets of 160GB. However, it was trained with more epochs and larger batch sizes compared to BERT, indicating higher computational resource requirements and longer training time (Acheampong et al., 2021).

To better handle long sequences, Longformer introduces a sparse attention mechanism to reduce computation (Beltagy et al., 2020). This allows each token to focus only on nearby tokens rather than the entire sequence. Unlike traditional models like BERT and RoBERTa, which can only process no more than 512 tokens, Longformer can handle up to 4096 tokens. It consistently achieves better performance than RoBERTa in downstream tasks involving long documents (Beltagy et al., 2020). The **Clinical-Longformer** model (Li et al., 2023) was fine-tuned for the clinical domain.

Table 1 summarises the **encoder-only models** used in our work and their corresponding fine-tuning datasets.

Model	Fine-tuned Dataset
Bio_Clinical BERT (Alsentzer et al., 2019)	MIMIC-III
medicalai/ClinicalBERT (Wang et al., 2023)	A large corpus of 1.2B words of diverse diseases
RoBERTa-base (Zhuang et al., 2021)	General Dataset (including BookCorpus, English Wikipedia, CC-News, OpenWebText, and Stories)
Clinical-Longformer (Li et al., 2023)	MIMIC-III

Table 1. Encoder-Only Models and Their Fine-tuned Datasets

2.3.3 Decoder-Only Models

In 2020, the performance of ChatGPT-3 (Brown et al., 2020) in question answering task caught researchers' attention to decoder-only architectures. As mentioned earlier, the Transformer decoder is an auto-regressive model. It can only refer to the synthesised words on the left side to generate the new word, without considering the context (which is called masked self-attention). This method made it more flexible in generating coherent text. Compared with BERT, the GPT series performed well in zero-sample and small-sample learning tasks by enlarging the size of the model. Even without fine-tuning, a simple prompt can help GPT generate a reasonable answer (Wu, 2024).

Unlike GPT, which improves models' performance by increasing dataset size and the number of parameters without limitations, Meta AI published a series of **Llama** Models. These models aim to maximise the use of limited resources - in other words, by extending training, they reduce the overall demand on computing resources. The latest Llama3 model requires only 8B to 70B parameters (AI, 2023), significantly less than GPT-3's 175 billion (Wu, 2024). Additionally, it outperforms GPT-3.5 Turbo in 5-shot learning (Context.ai, 2024).

2.3.4 Encoder-Decoder Models

T5 Family (Raffel et al., 2020a) is a classic example of the Encoder-Decoder Model. This architecture is particularly suitable for text generation tasks that require deep semantic understanding (Cai et al., 2022). T5 transforms all kinds of NLP tasks into a text-to-text format (Tsirmpas et al., 2024). Unlike BERT, which uses word-based masking and prediction, T5 processes text at the *fragment* level using "span corruption" to understand semantics (Tsirmpas et al., 2024). For the fill-in-the-blank task, instead of replacing the specific words with `<mask>` like BERT, T5 replaces the text fragments with an ordered set of `<extra_id_n>` to reassemble the long sequence text. T5 needs to pre-process the input text according to the tasks' requirements. A directive prefix should be added as a prompt.

Some language models fine-tuned with T5 on specific datasets, such as SciFive (fine-tuned in some science literature) (Phan et al., 2021) and ClinicalT5 (fine-tuned in clinical dataset MIMIC-III notes) (Lu et al., 2022), have shown excellent performance in their respective fields. The **T5 family models** used in this paper and their corresponding fine-tuned datasets are summarised in Table 2.

Model	**Pre-trained Dataset**	**Weight Initialisation Method**
T5-base (Raffel et al., 2020b)	Colossal Clean Crawled Corpus (C4)	Randomly Initialised
Clinical-T5-Base (Eric and Johnson, 2023; Goldberger et al., 2000)	MIMIC-III, MIMIC-IV	Initialised from T5-Base
Clinical-T5-Sci (Eric and Johnson, 2023; Goldberger et al., 2000)	PubMed Abstracts, PubMed Central	Initialised from SciFive
Clinical-T5-Scratch (Eric and Johnson, 2023; Goldberger et al., 2000)	MIMIC-III, MIMIC-IV	Randomly Initialised

Table 2. The T5 Family Models used in our work

2.3.5 Comparison and Limitations

According to (Cai et al., 2022), the encoder-decoder architecture performs best with sufficient training data. However, challenges in data collection can negatively affect its performance. Despite these challenges, different architectures are well-suited to different tasks. For example, for tasks requiring semantic understanding, such as text summarisation, the encoder-decoder architecture is the most effective. In contrast, for tasks that involve minor word modifications, the encoder-only structure works better. However, the decoder-only structure is not suitable for tasks with insufficient training data and long text processing, but performs well in few-shot question-answering tasks (Amin-Nejad et al., 2020b; Cai et al., 2022).

Following these discussions, Transformer-based Pre-trained Language Models (PLMs) have demonstrated strong performance in NLP tasks, but many challenges still remain.

2.4 Related Works on Clinical Text Generation

2.4.1 LT3: Label to Text Generation

LT3 (Belkadi et al., 2023) uses an encoder-decoder architecture to generate synthetic text from labels. As shown in Fig 2, labels such as medications are the input of the encoder, which can generate corresponding feature representations. The decoder generates prescription sequences based on these features. The pre-trained BERT tokenizer is used to split the input sequence into sub-words. LT3 is trained from scratch. Instead of using traditional greedy decoding, which may miss the global optimum, the authors proposed Beam Search Decoding with Backtracking (**B2SD**). This approach broadens the search range through a backtracking mechanism, preserving possible candidates for the optimal solution. To reduce time complexity, they used a probability difference function to avoid searching for low-probability words. Additionally, the algorithm penalises repeated sub-sequences and employs a logarithmic heuristic to guide the exploration of generation paths. The authors test LT3 on the 2018-n2c2 dataset, and evaluate the results using both quantitative metrics and downstream tasks. It was demonstrated that this model outperforms T5 in label-to-text generation. [3]

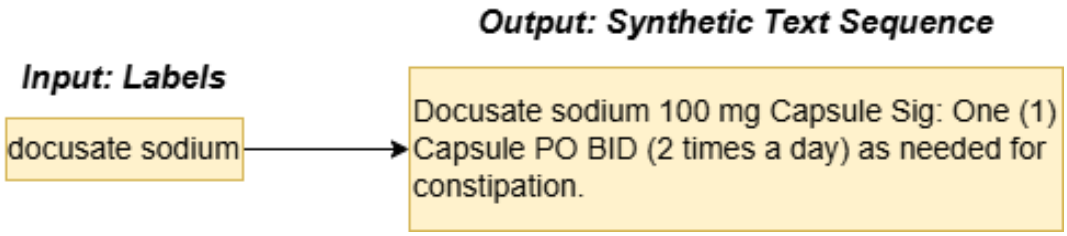

Figure 2. An Example of LT3 (Belkadi et al., 2023)

2.4.2 Seq2Seq Generation for Medical Dataset Augmentation

Amin-Nejad et al. (2020b) compare the performance of the Vanilla Transformer and GPT-2 using the MIMIC-III dataset in seq2seq tasks. Specifically, they input a series of structured patient information as conditions, as shown in Fig 3, to generate discharge summaries. They demonstrate that the augmented data outperforms the original data in downstream tasks (e.g. readmission prediction). Furthermore, they prove

[3] LT3 achieved significant improvements over the best-performing T5 model (T5 Base) in label-to-text generation, achieving improvements of up to 6.5 BLEU points and 0.02 in BERTScore. Unfortunately, when we tried applying B2SD to generate clinical letters, the results were somehow disappointing. This may be due to the length of clinical letters, B2SD consumes a lot of time on long text generation. Despite this, it still shows great potential in generating clinical data.

that Vanilla Transformer performs better with large samples, while GPT-2 excels in few-shot scenarios. However, GPT-2 is not suitable for augmenting long texts. Additionally, they used Bio_ClinicalBERT for the downstream tasks, and discovered that Bio_ClinicalBERT outperformed the baseline model (BERT) in almost all experiments. It suggests that Bio_ClinicalBERT can potentially replace BERT in the biomedical field. Interestingly, although the synthetic data have a low score on internal metrics (such as ROUGE and BLEU), the performance on downstream tasks is notably enhanced. This may be because augmenting text can effectively introduce noise into the original text, improving the model's generalisation to unseen data.

```
First ten tokens ... <H>
M <G>
65 <A>
white <E>
other pulmonary embolism and infarction | acute kidney failure ,
      unspecified | diarrhea | hypotension , unspecified <D>
other endoscopy of small intestine | gastroenterostomy without gastrectomy
      <P>
warfarin , 1mg Tablet | polysaccharide iron complex , 150MG | bisacodyl ,
      10MG SUPP | milk of magnesia , 30ML UDCUP <M>
blood culture : None | urine : staphylococcus species | mrsa screen : None
      | blood culture : None <T>
Calcium, Total , 10.0 , mg/dL | Bicarbonate , 25 , mEq/L | Hematocrit ,
      28.7 , \% , abnormal <L>
```

Figure 3. An Input Example of Conditional Text Generation (Amin-Nejad et al., 2020b)

According to their findings, decoder-only models like GPT-2 are not suitable for processing long text. Bio_ClinicalBERT is particularly effective for tasks in the clinical area, and Clinical Transformer is promising in augmenting medical data. This provides more possibilities for our task of generating synthetic clinical letters.

2.4.3 Discharge Summary Generation Using Clinical Guidelines and Human Evaluation Framework

Unlike the traditional supervised learning of fine-tuning language models (which requires a large amount of annotated data), (Ellershaw et al., 2024) generated 53 discharge summaries using only a one-shot example and a clinical guideline. Their research consists of two aspects: generating discharge summaries and a manual evaluation framework.

As shown in Fig 4, the authors used clinical notes from MIMIC-III as input, and incorporated a one-shot summary along with clinical guidance as prompts to generate discharge summaries by GPT-4-turbo. Initially, five sample synthetic summaries were evaluated by a clinician. Based on the feedback, the clinical guidance was revised to adapt to the generation task. Through iterative optimisation, the revised guidance, combined with the original one-shot sample, became the new prompt. Then the authors generated 53 discharge summaries using this method and invited 11 clinicians to do a final manual quantitative evaluation. Clinicians were invited to evaluate the error rate at the section level (e.g., Diagnoses, Social Context, etc). It includes four dimensions:

- Minor omissions;

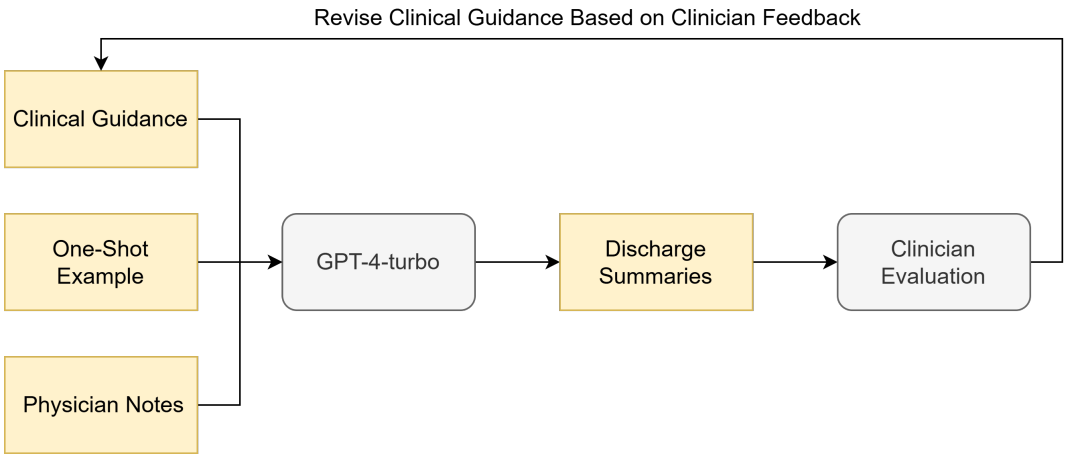

Figure 4. Workflow of Discharge Summary Generation Using Clinical Guidelines

- Severe omissions;
- Unnecessary text;
- Incorrect additional text.

Each discharge summary was evaluated by at least two clinicians, and the authors calculated agreement scores to evaluate the subjectivity during the human evaluation stage. Unfortunately, the inter-rater agreement was only 59.72%, raising concerns that the revised prompts based on such feedback might result in subjective synthetic summaries. Although this study partially addresses the issue of insufficient training data, and provides reliable human quantitative evaluation methods, it is still not well-suited for our investigation. Specifically, It required eleven clinicians to evaluate 53 synthetic samples, demonstrating the considerable time and manpower required. Therefore, there is still a long way to go before this technique can be used for large-scale text generation tasks.

2.4.4 Comparison of Masked and Causal Language Modelling for Text Generation

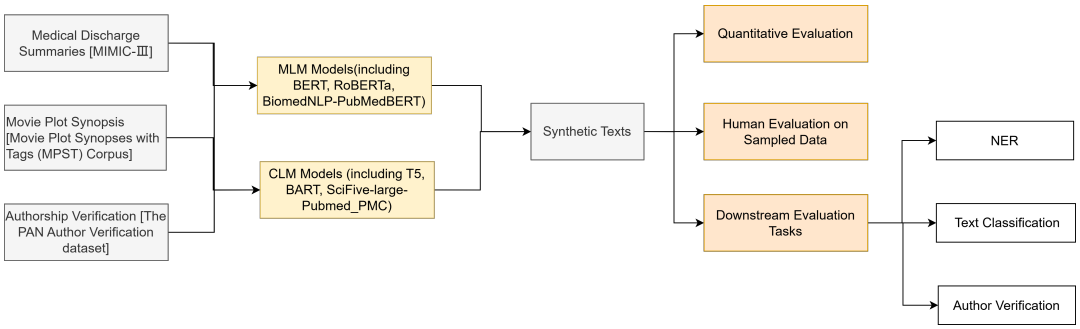

Figure 5. Workflow of MLM and CLM Comparison in Text Generation

Micheletti et al. (2024) compared masked language modelling (MLM, including BERT, RoBERTa, BiomedNLP-PubMedBERT) and causal language modelling (CLM, including T5, BART, SciFive-large-Pubmed_PMC) across various datasets in masking and text generation tasks. They used qualitative and quantitative evaluations, as well as downstream tasks, to assess the quality of the synthetic texts. Their workflow is shown in Fig 5. Based on these evaluations, the study yielded the following results:

- MLM models are better suited for text masking and generation tasks than CLM.
- Introducing domain-specific knowledge does not consistently improve model performance.
- Downstream tasks can adapt to the introduced noise. Although some synthetic texts might not achieve high quantitative evaluation scores, they can still perform well in downstream tasks. This matches the findings from Amin-Nejad (Amin-Nejad et al., 2020b).
- A lower random masking ratio (i.e., masked tokens / total tokens) can generate higher-quality synthetic texts.

These very recent findings provide insightful inspiration to our investigation. Our work will build on their research, expanding on masking strategies and focusing on the clinical domain. [4].

3 METHODOLOGIES AND EXPERIMENTAL DESIGN

Due to the sensitivity of clinical information, many clinical datasets are not accessible. As mentioned in Section 2, numerous studies use NLG techniques to generate clinical letters, and evaluate the feasibility of replacing the original raw clinical letters with synthetic letters. Most existing research involves fine-tuning PLMs or training Transformer-based models from scratch on their datasets through supervising learning. These studies explore different ways to learn the mapping from original raw text to synthetic text and work on generating synthetic data that are similar (or even identical) to the original ones. Our work, however, aims to find a method that can generate clinical letters that can *keep the original clinical story, while not exactly being the same as the original letters*. To achieve this objective, we employed various models and masking strategies to generate clinical letters. The experiment follows these steps:

1. **Data Collection and Pre-processing**: We first accessed clinical letter examples (A et al., 2000; Johnson et al., 2024a, 2023b) for an overview. The texts were segmented at the sentence level, and clinical entities as well as structural templates were extracted to capture the clinical narratives while maintaining clinical soundness.
2. **Randomly Masking**: We randomly masked the context and generated clinical letters by predicting masked tokens using different LLMs.
3. **Model Evaluation**: We evaluated synthetic letters generated by different language models. Based on their performance, we selected Bio_ClinicalBERT and worked on it.
4. **Masking Strategy Exploration**: We explored multiple masking strategies to retain clinical stories and diversity while removing private information. After generating clinical letters using these strategies, we evaluated their quality.
5. **Post-processing**: We applied post-processing techniques to further enhance the readability of synthetic letters.
6. **Downstream Task Evaluation**: We compared the performance of synthetic and original letters in a **downstream NER** task, to evaluate the usability of these synthetic letters.

An overall investigation workflow is shown in Fig 6.

[4] this work reports enriched findings based on our workshop paper (Ren et al., 2025)

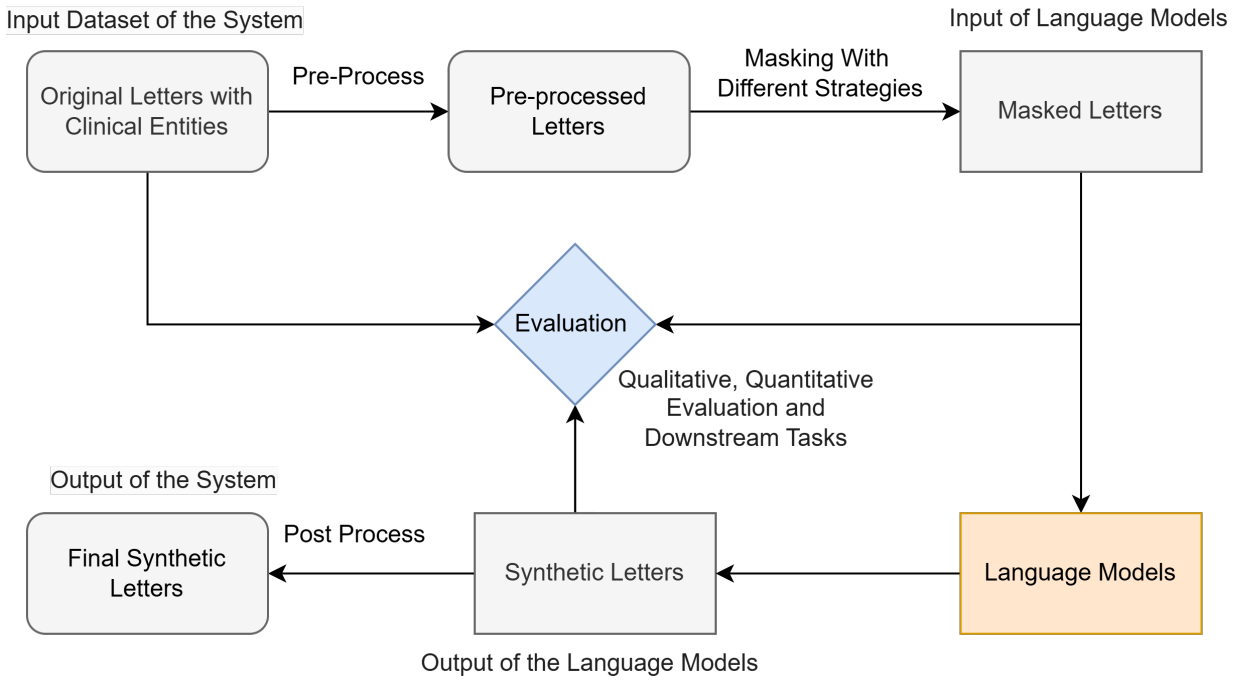

Figure 6. Overall Investigation Workflow for SYNTHETIC4HEALTH

3.1 Dataset

Based on the objective of this project, we need a dataset that includes both clinical **notes** and some clinical **entities**. The dataset we used is from the SNOMED CT Entity Linking Challenge (A et al., 2000; Johnson et al., 2024a, 2023b). It includes 204 clinical letters and 51,574 manually annotated clinical entities.

Clinical Letters The clinical letters are from a subset of discharge summaries in MIMIC-IV-Note (A et al., 2000; Johnson et al., 2023a). It uses clinical notes obtained from a healthcare system in the United States. These notes were de-identified by a hybrid method of the Rule-based Approach and Neural Networks. To avoid releasing sensitive data, the organisation also did a manual review of protected health information (PHI). In these letters, all PHI was replaced with three underscores '___'. The letters record the patient's hospitalisation information (including the reason for visiting, consultation process, allergy history, discharge instructions, etc.). They are saved in a comma-separated value (CSV) format file 'mimic-iv_notes_training_set.csv'. Each row of data represents an individual clinical letter. It consists of two columns, where the "note_id" column is a unique identifier for each patient's clinical letter, and the 'text' column contains the contents of the clinical letter. Since most language models have a limitation on the number of tokens to process (Sun and Iyyer, 2021), we tokenized the clinical letters into words using the 'NLTK' library and found that all clinical letters contained thousands of tokens. Therefore, it is necessary to split each clinical letter into multiple chunks to process them. These separated chunks should be merged in the end to generate the whole letter.

Annotated Clinical Entities The entities are manually annotated based on SNOMED CT. A total of 51,574 annotations cover 5,336 clinical concepts. They are saved in another CSV document which includes four columns: 'note_id', 'start', 'end', and 'concept_id'. The 'note_id' column corresponds to the 'note_id' in 'mimic-iv_notes_training_set.csv' file. The 'start' and 'end' columns indicate the position of annotated entities. The 'concept_id' can be used for entity linking with SNOMED CT. For example, for the

'note_id' '10807423-DS-19', , the annotated entity 'No Known Allergies' has a corresponding 'concept_id': '609328004'. This can be linked to SNOMED CT under the concept of 'Allergic disposition' (International, 2024).

An example of text excerpted from the original letter is shown in Figure 7. It contains the document structure and some free text. According to the dataset, document structure often corresponds to capital letters and colons ':'. Our primary goal is to mask the context that is neither part of the document structure nor annotated entities, and then generate a new letter, as both **structure** and clinical **entities** are essential for understanding clinical information (Meystre et al., 2014).

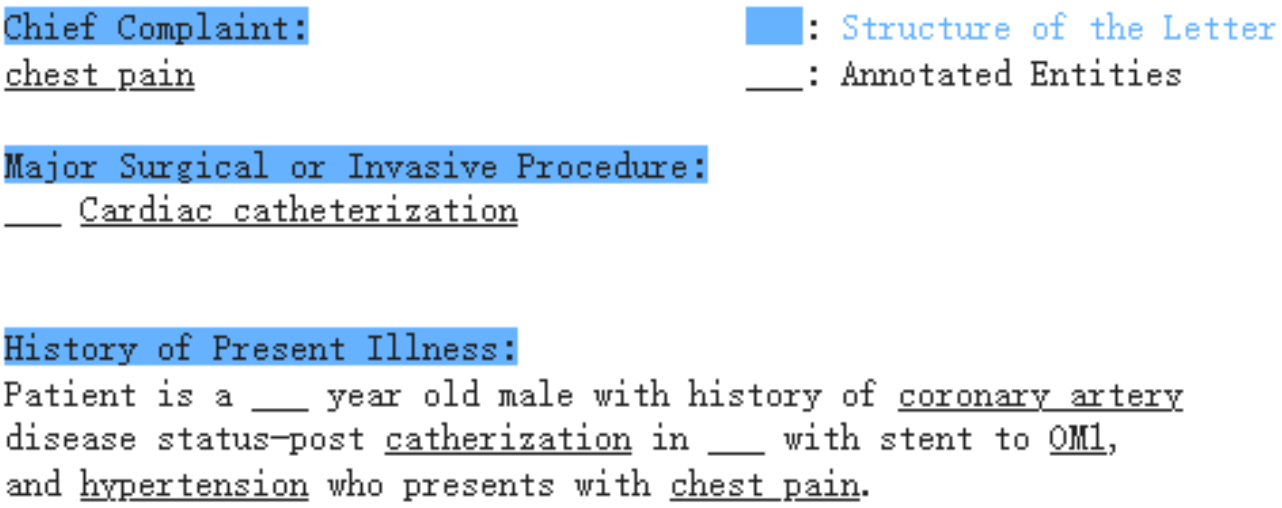

Figure 7. Text Excerpt from the Original Letter (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '17656866-DS-6')

3.2 Software and Environment

All codes and experiments in this paper are carried out in the integrated development environment (IDE) 'Google Colab Pro+' with a 52GB system RAM and 225GB disk space. The built-in T4 GPU (16GB VRAM) accelerates the inference process. The primary tools used in the paper include:

- **Programming Language and Environment:** Python `3.10` serves as the main programming language.
- **Deep Learning Framework:** PyTorch `2.3.1` is the core framework used for loading and applying pre-trained language models (PLMs).
- **Natural Language Processing Libraries:** This includes Hugging Face Transformers `4.42.4`, NLTK (Version $\geq$ `3.1`), and BERTScore `0.3.13`, etc. They are popular tools for text processing and evaluation in the NLP domain.
- **Auxiliary Tools:** Libraries such as pandas (Version $\geq$ `1.0.1`) and mpmath (`1.1.0` $\leq$ Version $<$ `1.4`) can support data management, mathematical operations, and other routine tasks.

3.3 Pre-Processing

The collected dataset involves different files and is entirely raw data. It is necessary to pre-process them before using them in generation tasks. The pre-processing of this system contains five steps: 'Merge dataset based on 'note_id'', 'Annotated Entity Recognition', 'Split Letters in Chunks', 'Word Tokenization' and 'Feature Extraction'. The pre-processing pipeline is shown in Fig 8.

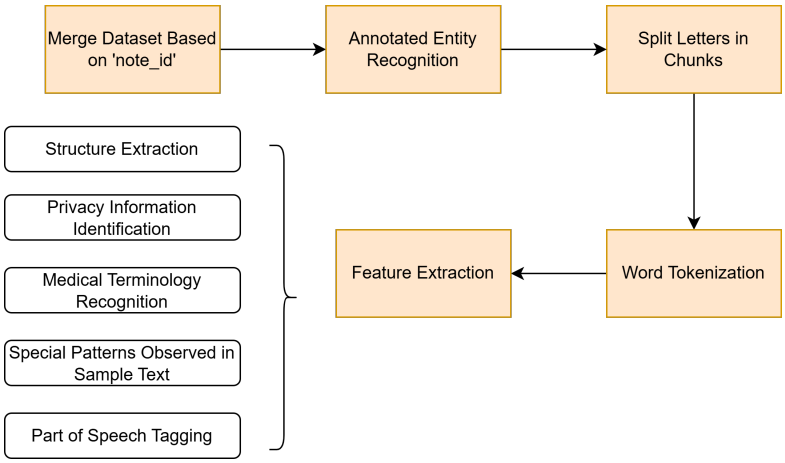

Figure 8. Pre-Processing Pipeline for SYNTHETIC4HEALTH

3.3.1 Merging Dataset and Annotated Entity Recognition

Initially, we merged the clinical letters file and annotations file into a new DataFrame. After this, we extracted manually annotated entities based on their index. An excerpt from an original letter is shown in Fig 9, and the manually annotated entities are listed in Table 3.

HISTORY OF PRESENT ILLNESS:
Patient is a ___ yo male previously healhty presenting w/ fall from 6 feet, from ladder. Patient landed on LLE w/ forced eversion and subsequent open fracture/dislocation. Denies head strike or LOC. Denies neck pain, back pain, chest pain, abd pain. Denies pelvic or thigh pain.

Was emergently reduced in ED under conscious sedation.

In the ED, initial vitals were 77 160/60 16 100%. Per the ED, the patient's exam did not show evidence of neurovascular symptoms.

Figure 9. Sample Text from Original Letters (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '10807423-DS-19')

3.3.2 Splitting Letters into Variable-Length Chunks

Typically, PLMs such as BERT, RoBERTa, and T5 have a limit on the number of input tokens, usually capped at 512 (Zeng et al., 2022). When dealing with text that exceeds this limit, common approaches

Entity	Start	End	Concept ID
fall	571	575	161898004
LLE	621	624	32153003
eversion	636	644	4196002
open fracture	660	673	397181002
dislocation	674	685	87642003
head strike	694	706	82271004
LOC	710	713	419045004
neck pain	722	731	81680005
back pain	733	742	161891005
chest pain	744	754	29857009
abd pain	756	765	21522001
pelvic	774	780	30473006
thigh pain	784	794	78514002
conscious sedation	833	851	314271007
vitals	874	880	118227000
neurovascular symptoms	963	986	308921004

Table 3. Extracted Entities and Their Details

include discarding the excess tokens or splitting the text into fixed-length chunks of 512 tokens. In addition, some studies evaluate the tokens' importance to decide which parts should be discarded (Hou et al., 2022).

In this work, each clinical letter ('note_id') contains thousands of tokens, as mentioned in Section 3.1, to preserve as much critical clinical information as possible, we avoided simply discarding tokens. Instead, we adopted a splitting strategy based on **semantics**. Each block is not a fixed length. Rather, they are complete **paragraphs** that are as close as possible to the token limit. This approach aims to help the model better capture the meaning and structure of clinical letters, thereby improving its ability to retain essential clinical information while efficiently processing the text. In fact, we initially generated letters at the sentence level. However, it was found that processing at the sentence level is not only time-consuming, but also fails to provide the model with enough information for inference and prediction. This is why the letters are processed in chunks rather than in sentences.

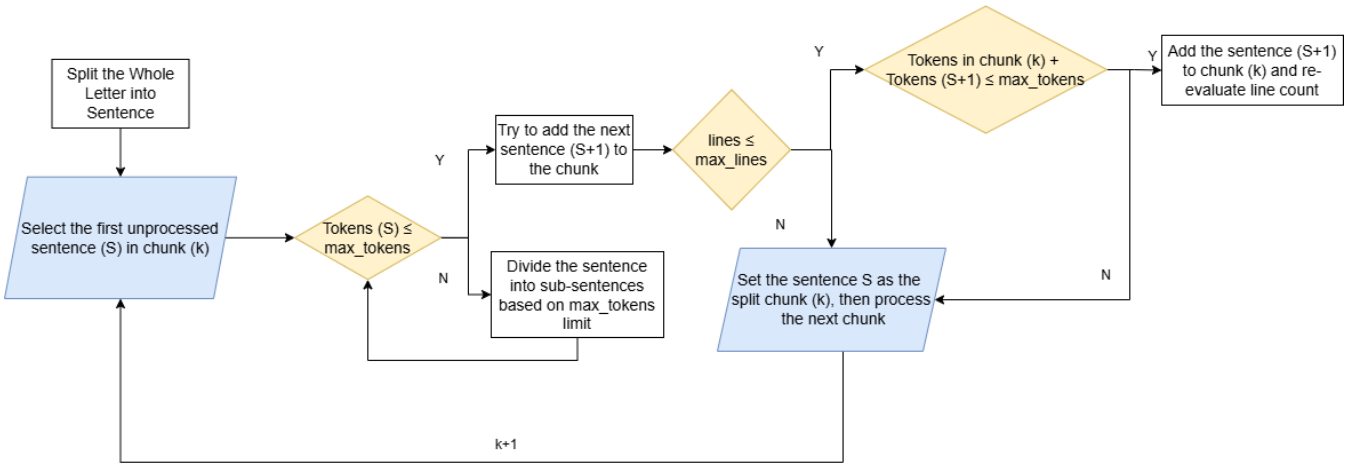

Figure 10. Text Chunking Workflow for SYNTHETIC4HEALTH

As shown in Fig 10, each raw letter is split into sentences first. We used the pre-trained models provided by the 'NLTK' library, which combines statistical and machine-learning approaches to identify sentence

boundaries. Each clinical letter is treated as a separate processing unit, with the first sentence automatically assigned to the first text block (chunk). To control the length of each chunk, we set a **maximum line count** parameter (max_lines). If the first sentence already meets the value of 'max_lines', the chunk will only contain this one sentence. Otherwise, subsequent sentences will be added to the chunk until the line count up to the max_lines.

Extra care is needed when handling text with specific formats, such as medication dosage descriptions, as shown in Fig 11. Because there is no clear sentence boundary, these sentences may exceed the tokens limitation. To address this, we first check whether the sentence being processed exceeds the token limit (max_tokens). If it does not, the sentence will be added to the current chunk. Otherwise, the sentence should be split into smaller chunks, each no longer than 'max_tokens'. This operation helps **balance processing efficiency** while maintaining semantic integrity. In the example shown in 11, although using line breaks to split the text seems to be more flexible, considering time complexity and the requirement to index the annotated entities, this method was not chosen.

```
Discharge Medications:
1. Acetaminophen 1000 mg PO Q8H
2. Docusate Sodium 100 mg PO BID
3. Enoxaparin Sodium 40 mg SC QHS
Start: Today - ___, First Dose: Next Routine Administration
Time
RX *enoxaparin 40 mg/0.4 mL 1 syringe at bedtime Disp #*14
Syringe Refills:*0
4. OxycoDONE (Immediate Release)  ___ mg PO Q4H:PRN pain
RX *oxycodone 5 mg ___ tablet(s) by mouth q3hrs Disp #*80 Tablet
Refills:*0
```

Figure 11. Sentence Fragment Exceeding Token Limit (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '10807423-DS-19')

3.3.3 Word Tokenisation

To prepare the text for model processing, we split each chunk of text into smaller units: tokens. The tokenisation methods can be categorised into two types: one for feature extraction and the others for masking and generation.

For the tokenization aimed at feature extraction, we used the 'word_tokenize' method from the 'NLTK' library. It is helpful to preserve the original features of the words, which is especially important for retaining clinical entities. For instance, in the sentence "Patient is a ___ yo male previously healthy presenting w/ fall from 6 feet, from ladder." Word boundaries such as spaces can be automatically detected for tokenization. The results of different tokenization methods are shown in the Table 4.

As for the tokenization used for masking and generating, we retained the original models' tokenization methods. The specific tokenization approach varies by model, as shown in Table 4. For example, BERT family models use Word-Piece tokenization, which initially splits text by spaces and then further divides the words into sub-words (Zhuang et al., 2021). This approach is particularly effective for handling words that are not in the pre-training vocabulary and is especially useful for predicting masked words. For complex clinical terms, however, these models rely heavily on a predefined dictionary, which can result in unsatisfactory tokenization and hinder the model's understanding. For instance, the word 'COVID-19' is

tokenized by BERT into [‘co’, ‘##vid’, ‘-’, ‘19’]. In contrast, the T5 family models use Sentence-Piece tokenization. It does not rely on space to split the text. Instead, this method tokenizes directly from the raw text, making it better suited for handling abbreviations and non-standard characters (e.g. ‘COVID-19’), which are common in clinical letters.

It is important to note that although all BERT family models use Word-Piece tokenization, the results can still differ. This is because different models use different vocabularies during pre-training, leading to variations in tokenization granularity. The tokenization methods for each model are detailed in Table 4. Each tokenization approach has its own advantages and disadvantages for processing clinical letters. Therefore, exploring how these models impact the clinical letter generation is also a requirement of our project.

Operation / Model	**Tokenization Method**	**Tokenized Output**
Feature Extraction	Word Tokenization	[‘Patient’, ‘is’, ‘a’, ‘___’, ‘yo’, ‘male’, ‘previously’, ‘healthy’, ‘presenting’, ‘w’, ‘fall’, ‘from’, ‘6’, ‘feet’, ‘,’, ‘from’, ‘ladder’, ‘.]
medicalai / ClinicalBERT	Subword-Enhanced Word-Piece	[‘patient’, ‘is’, ‘a’, ‘_’, ‘_’, ‘_’, ‘yo’, ‘male’, ‘previously’, ‘healthy’, ‘presenting’, ‘w’, ‘/’, ‘fall’, ‘from’, ‘6’, ‘feet’, ‘,’, ‘from’, ‘la’, ‘##dder’, ‘.]
BERT-base, Bio_ClinicalBERT	Standard Word-Piece	['patient', 'is', 'a', '_', '_', '_', 'yo', 'male', 'previously', 'healthy', 'presenting', 'w', '/', 'fall', 'from', '6', 'feet', ',', 'from', 'ladder', '.']
Clinical-Longformer, RoBERTa	Detailed Word-Piece	['Pat', 'ient', 'Ġis', 'Ġa', 'Ġ___', 'Ġyo', 'Ġmale', 'Ġpreviously', 'Ġhealthy', 'Ġpresenting', 'Ġw', '/', 'Ġfall', 'Ġfrom', 'Ġ6', 'Ġfeet', ',', 'Ġfrom', 'Ġladder', '.']
T5 Family (T5 base, SCI T5, Clinical T5)	Sentence-Piece	[‘__Patient’, ‘__is’, ‘__’, ‘a’, ‘__’, ‘_’, ‘_’, ‘_’, ‘__’, ‘y’, ‘o’, ‘__male’, ‘__previously’, ‘__healthy’, ‘__’, ‘presenting’, ‘__’, ‘w’, ‘/’, ‘__fall’, ‘__from’, ‘__6’, ‘__feet’, ‘,’, ‘__from’, ‘__ladder’, ‘.’]

Table 4. Comparison of Tokenization Methods for Different LMs on Sentence
”Patient is a ___ yo male previously healthy presenting w/ fall from 6 feet, from ladder.”

3.3.4 Feature Extraction

Since we aim to generate de-identified clinical letters that can preserve clinical narratives during masking and generation, it is necessary to extract certain features beforehand. We extracted the following features, with an example provided in Figure 12 and Table 5.

- **Document Structure:** This feature is identified by a rule-based approach. As mentioned in Subsection 3.1, structural elements (or templates) often correspond to the use of colons ‘:’. They should not be masked to preserve the clinical context.
- **Privacy Information Identification:** In this part, we used a hybrid approach. To identify sensitive information such as ‘Name’, ‘Date’, and ‘Location (LOC)’, We employed a NER toolkit from Stanza (Qi et al., 2020). To handle privacy information like phone numbers, postal codes, and e-mail addresses, We implemented a rule-based approach. Specifically, We devised several regular expressions to match the common formats of these data types. These identified privacy information should be masked.

- **Medical Terminology Recognition:** A NER toolkit pre-trained on the dataset i2b2 is used here (Zhang et al., 2021). It can identify terms like 'Test', 'Treatment', and 'Problem' in free text. Although our dataset has already been manually annotated, these identified terms can serve as a supplement to the pre-annotated terms.
- **Special Patterns Observed in Sample Text:** Some specific patterns, like medication dosages (e.g. enoxaparin 40 mg/0.4 mL) or special notations (e.g. 'b.i.d.'), may carry significant meaning. We retained these terms unless they were identified as private information to preserve the clinical background of the raw letters.
- **Part of Speech (POS) Tagging:** Different parts of speech (POS) play distinct roles in interpreting clinical texts. We aim to explore how these POS influence the model's understanding of clinical text. To achieve this, We used a toolkit (Zhang et al., 2021) trained on the MIMIC-III (Johnson et al., 2016) dataset for POS tagging. It performs better than SpaCy [5] and NLTK in handling clinical letters.

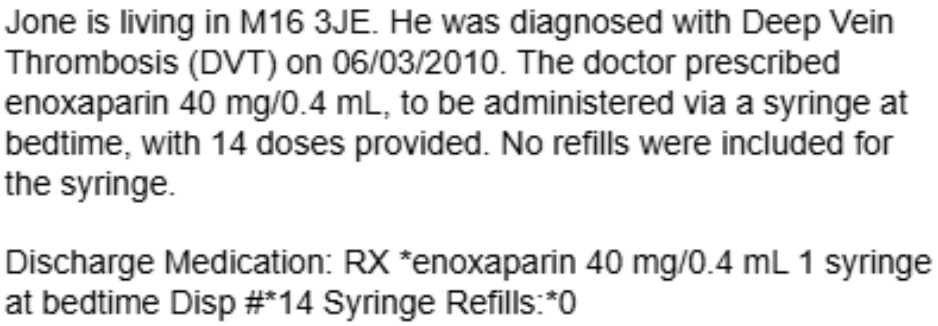

Figure 12. Example Sentence for Feature Extraction (See Table 5 for Extracted Features)

3.4 Clinical Letter Generation

We discuss the models and masking strategies that are used in generating synthetic clinical letters. It is important to clarify that our key objective is to generate letters that differ from the original ones, rather than being exact copies, as the same statement may indirectly reveal the patients' privacy. Although fine-tuning the model can always improve precision and enhance the model's semantic comprehension ability, it tends to produce letters that are too closely aligned with the originals. This also causes the fine-tuned model to rely too heavily on the original dataset, compromising its ability to generalise. Therefore, simply fine-tuning the model is not ideal if the PLMs can already generate the readable text. Instead, we should concentrate on how to *protect clinical terms and patient narratives as well as avoid privacy breaches*.

As discussed in Section 2.3 and Section 2.4, decoder-only models struggle with processing long texts that require contextual understanding (Amin-Nejad et al., 2020b). Additionally, deploying them requires substantial computing resources and time. Therefore, we explored various pre-trained language models (PLMs), including both encoder-only and encoder-decoder models in this paper. After evaluating their ability to generate synthetic letters from our dataset, we focused on **Bio_ClinicalBERT** - a well-performed model in our task, to experiment with different masking strategies. Additionally, from the discussion in

[5] https://spacy.io/

Feature Extraction Operation	**Extracted Features**
Structure Extraction	Discharge Medication:
Privacy Information Identification	Jone (PERSON) 06/03/2010 (DATE) Postal Code: M16 3JE
Medical Terminology Recognition	Deep Vein Thrombosis (PROBLEM) DVT (PROBLEM) enoxaparin (TREATMENT) the syringe (TREATMENT) RX enoxaparin (TREATMENT)
Special Patterns Observed in Sample Text	'40 mg/0.4 mL' (122, 134) '40 mg/0.4 mL ' (284, 297) '#14' (323, 327) '0' (344, 346)
POS Tagging	'Jone', 'PROPN' 'is', 'AUX' 'living', 'VERB' 'in', 'ADP' *(Partial List)*

Table 5. Example: Summary of Feature Extraction Operations and Extracted Features

Section 3.3, we need to split the text into various-length-chunks. So the appropriate *length of these chunks* is also experimented with Bio_ClinicalBERT.

3.4.1 Encoder-Only Models with Random Masking

As mentioned earlier, the primary method for this paper involves masking and generation. We focused extensively on encoder-only models because of their advantage in bi-directional semantic comprehension. These encoder-only models, including BERT, RoBERTa, and Longformer—detailed in Section 2.3—were compared for their performance. Given the clinical focus of this task, we particularly explored model variants that were fine-tuned on clinical or biological datasets. However, as no clinically fine-tuned RoBERTa (Zhuang et al., 2021) variant was available, the **RoBERTa-base** was used for comparisons. Specifically, the encoder-only models we explored include Bio_ClinicalBERT (Alsentzer et al., 2019), medicalai/ClinicalBERT (Wang et al., 2023), RoBERTa-base (Zhuang et al., 2021), and Clinical-Longformer (Li et al., 2023).

We used the standard procedure for Masked Language Modelling (**MLM**). First, the tokens that need to be masked are selected. They are then corrupted, resulting in masked text - that includes both masked and unmasked tokens. Next, the model predicts the masked tokens and replaces them with the ones that have the highest probabilities.

3.4.2 Encoder-Decoder Models with Random Masking

Although encoder-decoder models are not typically used for masked language modelling, they are well-suited for text generation. The architecture of T5, in particular, is designed to maintain the coherence of the text (Raffel et al., 2020a). Therefore, we included the T5 family models in comparisons.

The process of generating synthetic letters with encoder-decoder models is very similar to that with encoder-only models. The difference is that, unlike the BERT family, which automatically masks tokens and replaces them with '`<mask>`', the T5 family models *do not have any built-in masking function*. As a result, we identified the words that needed to be masked by **index** and removed them, which are

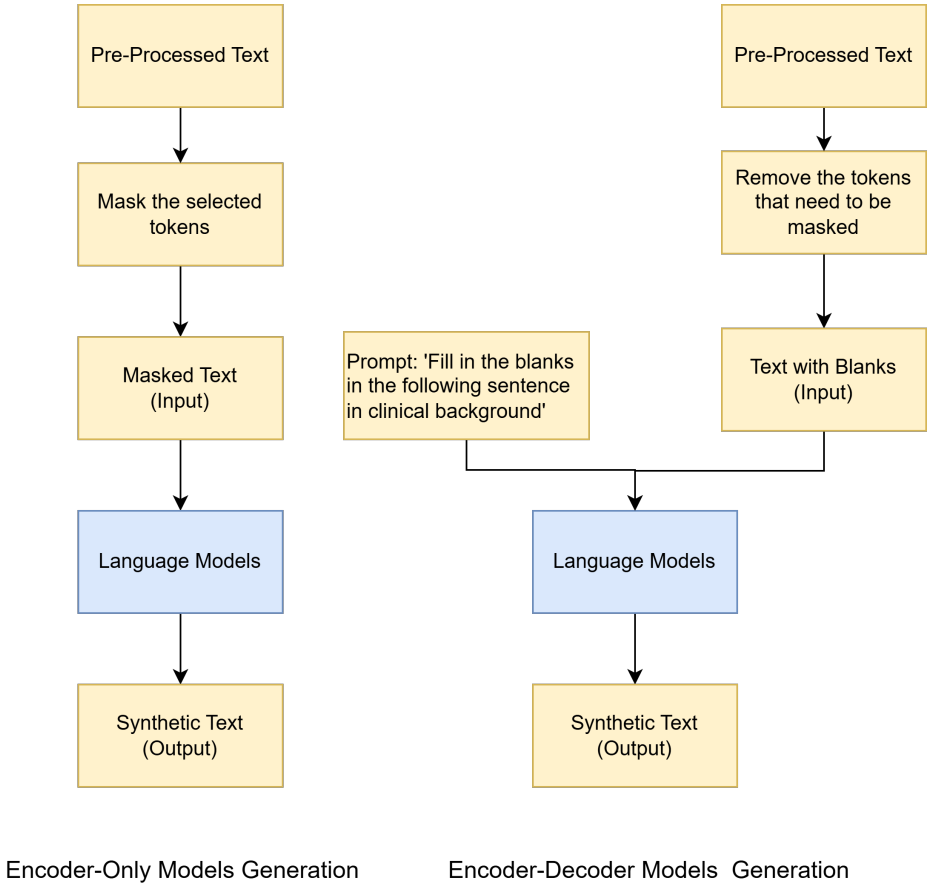

Figure 13. SYNTHETIC4HEALTH: Comparison of Encoder-Only and Encoder-Decoder Model Architectures

represented as 'extra_id_x' in the T5 family models. The text, with these words removed, was then used for the generation, which we refer to as '**text with blanks**'. To maintain consistency in the format, we later replaced 'extra_id_x' with '<mask>' when displaying the masked text. Additionally, the T5 family models **require a prompt** as part of the input. For this task, the complete input was structured as 'Fill in the blanks in the following sentence in the clinical background' + 'text with blanks'. In this paper, we used T5-base (Raffel et al., 2020b), Clinical-T5-Base (Eric and Johnson, 2023; Goldberger et al., 2000), Clinical-T5-Sci (Eric and Johnson, 2023; Goldberger et al., 2000), and Clinical-T5-Scratch (Eric and Johnson, 2023; Goldberger et al., 2000) for comparison. The comparison of encoder-only and encoder-decoder model architectures is shown in Fig 13.

3.4.3 Different Masking Strategies with Bio_ClinicalBERT

To make the synthetic letters more readable, clinically sound, and privacy-protective, different masking strategies are tested based on the following principles.

1. **Preserve Annotated Entities:** The manually annotated entities should not be masked to retain the clinical knowledge and context.
2. **Preserve Extracted Structures:** Tokens that are part of the document structure should be preserved as templates for clinical letters.

3. **Mask Detected Private Information:** This is helpful in de-identification. Although the dataset we use is de-identified, this approach may be useful when this system is deployed with real-world data.
4. **Preserve Medical Terminology:** It still aims to retain clinical knowledge, as some diseases and treatments were not manually annotated.
5. **Preserve Non-Private Numbers:** Certain numbers, such as drug dosage or heart rates, are indispensable for clinical diagnosis and treatment. However, only non-private numbers should be retained, while private information (such as phone numbers, ages, postal codes, dates, and email addresses) should be masked.
6. **Preserve Punctuation:** Punctuation marks such as periods ('.') and underscores ('___') should not be masked, as they clarify the sentence boundaries and make the synthetic letters more coherent (Lamprou et al., 2022).
7. **Retain Special Patterns in Samples:** Tokens that match specific patterns (e.g. 'Vitamin C ^1000 mg', 'Ibuprofen $>$ 200 mg', etc) should be retained, as they may contain important clinical details. These patterns are summarised by analysing raw sample letters.

Based on the principles above, different masking strategies were experimented with:

1. **Mask Randomly:** Tokens that can be masked are selected randomly from the text. We experimented with *masking ratios* ranging from 0% to 100% in 10% increments. This approach helps to understand how the number of masked tokens influences the quality of synthetic letters, and provides a baseline for other masking strategies.
2. **Mask Based on POS Tagging:** We experimented with different configurations in this section, such as masking only **nouns**, only **verbs**, etc. It is helpful to understand how POS influences the models' context understanding. Similar to the random masking approach, We selected the tokens based on their POS configuration and masked them in 10% increments from 0% to 100%.
3. **Mask Stopwords:** Stopwords generally contribute little to the text's main idea. Masking stopwords serves two purposes: reducing the *noise* for model understanding and increasing the *variety* of synthetic text by predicting these words. Moreover, they do not influence crucial clinical information. This approach is highly similar to the one used in 'Mask Based on POS Tagging'. The only difference is the criteria for selecting tokens. Specifically, tokens are selected based on whether they are stopwords rather than on their POS. '**NLTK**' library is used for detecting stopwords in the text.
4. **Hybrid Masking Using Different Ratio Settings:** After employing the aforementioned masking strategies, we observed the influence of these elements. Additionally, we experimented with their *combinations* at different masking ratios based on the outcomes, such as masking 50% nouns and 50% stopwords simultaneously.

3.4.4 Determining Variable-Length-Chunk Size with Bio_ClinicalBERT

As mentioned in Section 3.3, we utilise two parameters in our chunk segment procedure: 'max_lines' and 'max_tokens'. 'max_lines' represents the desired length of each chunk, while 'max_tokens' is related to the computing resources and model limitations. These two parameters determine the final length of each chunk together. Although most models we used have a limit of 512 tokens (except for the Longformer, which can process up to 4096 tokens), we set 256 as the value for **'max_tokens'** due to the computing resources constraints.

As for 'max_lines', we experimented with values starting from 10 lines, increasing by 10 lines each time, and calculated the average tokens for each chunk. Once the token growth began to slow, we refined the search by using **finer increments**. Finally, we selected the number of lines at which the average tokens per chunk stopped growing. This is because more lines in each chunk provide more information for the model to predict masked tokens. However, if the chunk length reaches a critical threshold, it indicates that the primary limitation is 'max_tokens', not 'max_lines'. Continuing to increase 'max_lines' would lead to additional computational overhead, as the system would have to repeatedly check whether adding the next sentence meets the required line count.

3.5 Evaluation Methods

Both quantitative and qualitative methods will be used to evaluate the performance. Additionally, a downstream task (NER) is employed to assess whether the synthetic clinical letters can replace the original raw data. The evaluation methods pipeline is illustrated in Fig 14.

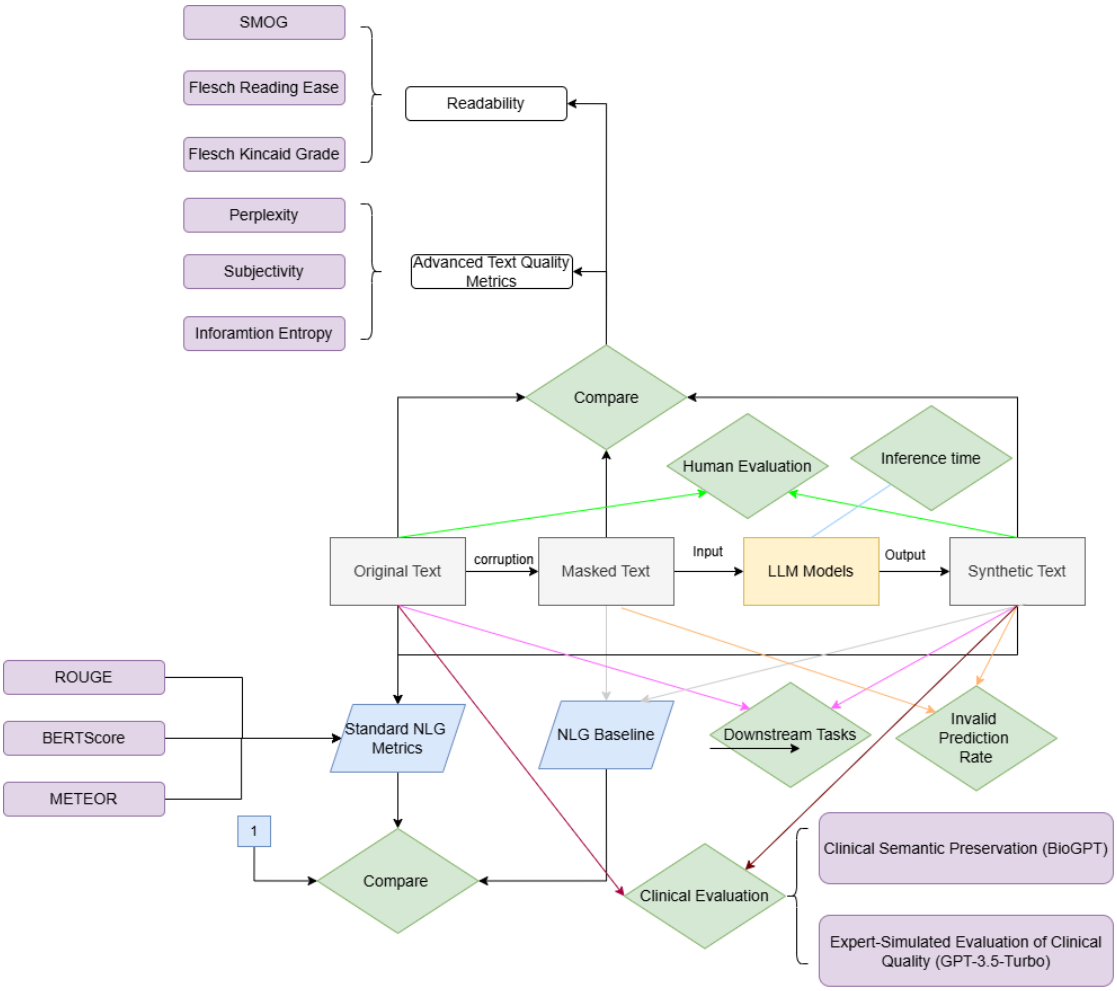

Figure 14. SYNTHETIC4HEALTH Evaluation Pipeline

3.5.1 Quantitative Evaluation

To comprehensively evaluate the quality of the synthetic letters, we used quantitative evaluation from multiple dimensions, including the model's **inference** performance, the **readability** of the synthetic letters, and their **similarity** to the raw data. The specific metrics are listed below.

Standard NLG Metrics It covers standard NLG evaluation methods such as ROUGE, BERTScore, and METEOR. ROUGE measures literal similarity, BERTScore evaluates semantic similarity, and METEOR builds on ROUGE by taking synonyms and word order into account. It provides a more comprehensive evaluation of synthetic text (Banerjee and Lavie, 2005).

These evaluations will be performed by comparing synthetic text with the original text. Moreover, a baseline is calculated by comparing masked text to the original text. The evaluation score should exceed the baseline but remain below '1', ensuring it does not exactly replicate the original text.

Readability Metrics To evaluate the readability, we calculated **SMOG**, **Flesch Reading Ease**, and **Flesch-Kincaid Grade Level**. Given our clinical focus, we prioritise SMOG as the primary readability metric, with Flesch Reading Ease and Flesch-Kincaid Grade Level as reference standards. In this analysis, we will compare the readability metrics of the synthetic text with those of the original and masked texts. The evaluation results should closely approximate the original text's metrics. Significant differences (DuBay, 2004)[6] may suggest that the model cannot preserve semantic coherence and readability adequately.

Advanced Text Quality Metrics In this part, we calculated the **perplexity**, **subjectivity**, and **information entropy**. We want the synthetic letters to be useful in training clinical models. Therefore, perplexity should not be far away from the value of the original letters. As for subjectivity and information entropy, we expect the synthetic letters to be both subjective and informative.

Invalid Prediction Rate We calculated the invalid prediction rate for each generation configuration. This ratio is determined by dividing the number of invalid predictions (such as punctuation marks or subwords) by the total number of masked words that need to be predicted. We expect the model to generate more meaningful words. Since punctuation marks are not masked, the model should avoid generating too many non-words. This metric can provide insights into the model's inference capability.

Inference Time The inference time for each generation configuration across the whole dataset (204 clinical letters) was recorded. Shorter inference times indicate lower computational resource consumption. When this system is deployed on large datasets, it is expected to save both time and computing resources.

3.5.2 Qualitative Evaluation

In the quantitative evaluation, we not only calculated the evaluation metrics for the entire dataset, but also recorded the results for each individual synthetic clinical letter. Interestingly, while some synthetic texts exhibited strong performance according to most metrics, they did not always appear satisfactory upon "visual" inspection. Conversely, some synthetic letters with average metrics may appear more visually appealing.

Although human evaluation is the most reliable approach for evaluating clinical letters, it is limited by availability and cost. Therefore, combining qualitative and quantitative evaluations helps in identifying suitable quantitative metrics for assessing our model's performance. Once identified, one of these metrics can be used as the **primary standard**, while the others serve as supporting indicators. As a workaround,

[6] We define a significant difference as a change of 1 SMOG grade, 1 Flesch-Kincaid Grade Level, or 10 points in Flesch Reading Ease, as these thresholds approximately correspond to a shift of one grade level or readability tier.

we selected a small sample of representative clinical letters based on the evaluation results. Subsequently, we reviewed the outcomes to better understand how different generation methods impacted these results, while also evaluating their correspondence with the quantitative metrics.

3.5.3 Downstream NER Task

Beyond qualitative and quantitative evaluation, we can also apply synthetic clinical letters in a downstream NER task. This is helpful to further evaluate their quality and their potential to replace original ones in clinical research and model training.

ScispaCy [7] and spaCy [8] are used in this part. As shown in Fig 15, they extract features from the text and learn the weights of each feature through neural networks. These weights are updated by comparing the loss between the predicted probabilities and actual labels. If a word does not belong to any label, it is classified as 'O' (outside any entity). SpaCy initialises these weights randomly. However, the version of ScispaCy we use, ('en_ner_bc5cdr_md'), is specifically fine-tuned on the **BC5CDR** corpus. It focuses more on the entities 'chemical' and 'disease' while retaining the original general features.

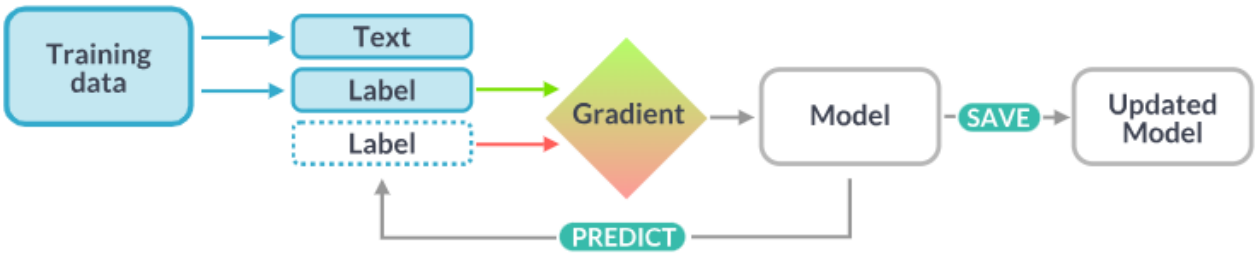

Figure 15. Training Process for spaCy NER Model [9]

In this downstream NER task, as shown in Fig 16, we initially extracted entities from letters using ScispaCy. Subsequently, these entities were used to train a base spaCy model. The trained model was then employed to extract entities from the testing set. Finally, we can compare these newly extracted entities with those originally extracted by ScispaCy, and the evaluation scores can be calculated. These steps were performed on both original clinical letters and synthetic letters, to assess whether the synthetic letters can potentially replace the original ones.

3.5.4 Clinical Evaluation

Clinical Semantic Preservation: To evaluate how much clinical information is preserved, we use BioBERT (Luo et al., 2022) for a rough estimate. Specifically, we tokenize both the original and synthetic letters, obtain their embeddings using BioBERT, and compute the cosine similarity between them. Since BioBERT is trained on biomedical corpora, its embeddings are expected to capture clinical semantic features. A high similarity score indicates that clinical information is largely preserved. However, it is important to note that this method only evaluates the effectiveness of preserving clinical narratives at the semantic level and does not guarantee medical factuality.

[7] https://allenai.github.io/scispacy/

[8] https://spacy.io/

[9] Image Credit https://spacy.io/usage/training

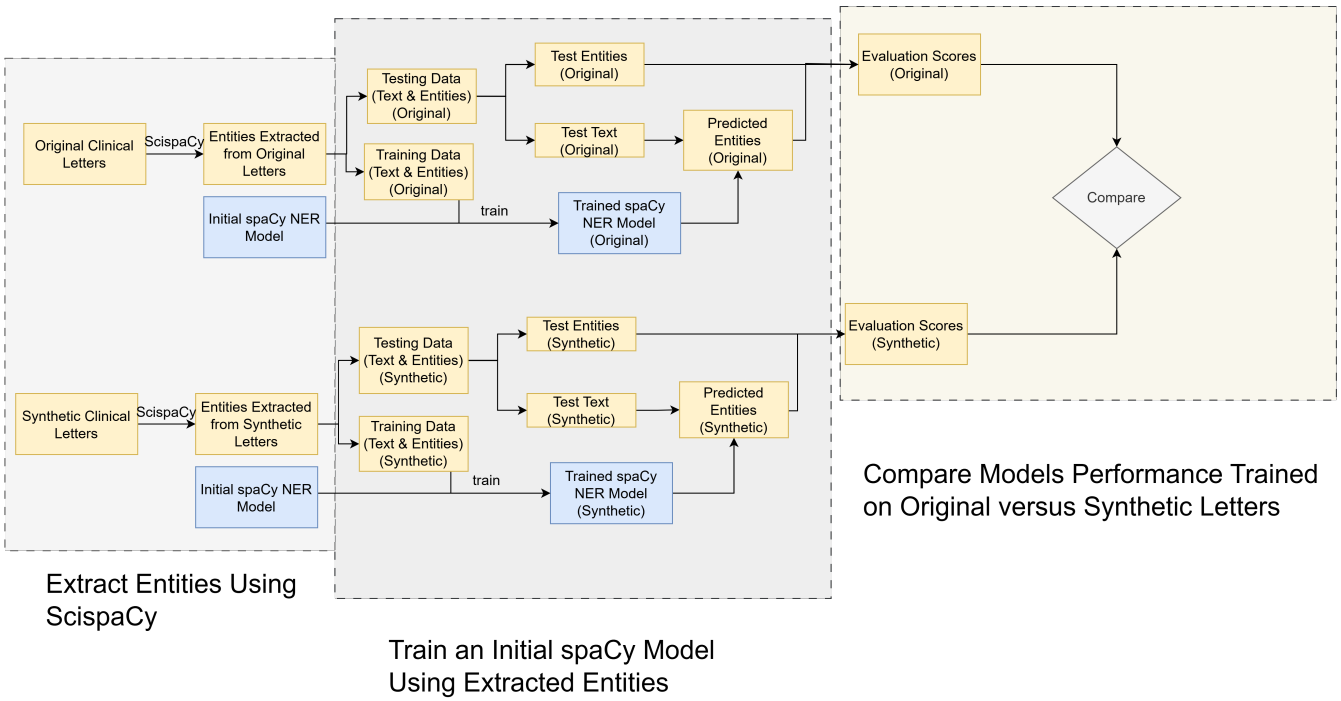

Figure 16. SYNTHETIC4HEALTH Workflow of Downstream NER Task

Expert-Simulated Evaluation of Clinical Quality: To further evaluate the clinical usefulness of our synthetic letters, we employ GPT-3.5-Turbo (OpenAI, 2023) through prompt-based evaluation. Specifically, we evaluate the results from two perspectives: **clinical soundness** and **narrative coherence**. Clinical soundness measures whether the content aligns with medical factuality, while narrative coherence evaluates whether the letter is contextually consistent and resembles a real-world clinical letter. The prompt we used is shown in Fig 17

```
Suppose you are a senior clinical doctor, you need to evaluate the quality of
clinical letters.

Please rate the following clinical letter from 0 to 1 on these two dimensions:

1. Clinical Soundness - Is the diagnosis and treatment medically reasonable?
2. Narrative Coherence - Does it flow logically like a real clinical note?

Use this format:
Clinical Soundness: <score> - <brief explanation>
Narrative Coherence: <score> - <brief explanation>

---

Letter:
\"\"\"{letter}\"\"\"
```

Figure 17. GPT-3.5-turbo Prompt for Clinical Evaluation

3.6 Post-Processing

3.6.1 Filling in the blanks

As described in Section 3, the dataset we used has been de-identified. All private information is replaced by three underscores '___'. We hope that the synthetic clinical letters can maintain a certain degree of clinical integrity without revealing any private patient information. To address this, a post-processing step was added to the synthetic results. This process involves masking the three underscores ('___') detected and using PLMs to predict the masked part again. For example, if the original text is '___ caught a cold'. The post-processing result should ideally be 'John caught a cold' or 'Patient caught a cold'. Such synthetic clinical letters can better support clinical model training and teaching.

In this part, we used Bio_ClinicalBERT and BERT-base models. Although Bio_ClinicalBERT is better at clinical information understanding, this issue is not directly related to clinical practice, so we used BERT-base for comparison.

3.6.2 Spelling Correction

Since our data comes from real-world sources, it is inevitable that some words may be misspelled by doctors. These spelling errors can negatively impact the model's training process or hinder clinical practitioners' understanding of the synthetic clinical letters. Although some errors are masked and re-generated, our masking ratio is not always 100%, so some incorrect words may still exist. A toolkit 'TextBlob' (Loria, 2024) is added to correct these errors. Specifically, it uses a rule-based approach that relies on a **built-in vocabulary library** to detect and correct misspellings.

3.7 Summary

In this section, we introduced the experimental design and subsequent implementation steps: from project requirement, data collection to environmental setup, pre-processing, masking and generating, post-processing, downstream NER task, clinical evaluation, and both qualitative and quantitative evaluation. An example of the entire process flow is shown in Figure 18.

4 EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Chunk Segmentation Effects on Inference Time

As mentioned in Subsection 3.4.4, we set 'max_lines' as a variable and the 'max_tokens' equal to 256. A series of increasing 'max_lines' were tested until the average tokens per chunk peaked. We initially did this on a small sample (7 letters). The results are shown in Table 6 for Bio_ClinicalBERT model.

We can see that the average tokens per chunk reach a peak as the 'max_lines' parameter increases to 41. Similarly, inference time decreases as 'max_lines' increases to 41, but it rises again once it exceeds this value. This experiment was also conducted on slightly larger samples of 10 and 30 letters. All of them showed the same trend. However, the inference time here may only reflect an overall trend, not exact results, as it is influenced by many factors, not only the chunk size but also the internet speed, etc.

4.2 Random Masking: Qualitative Results

We employed both the encoder-only and encoder-decoder models to mask and generate the data, yielding numerous interesting results for human evaluation. Given space constraints, only a simple example is provided here. Following the masking principles in Section 3.4, the eligible tokens were randomly selected

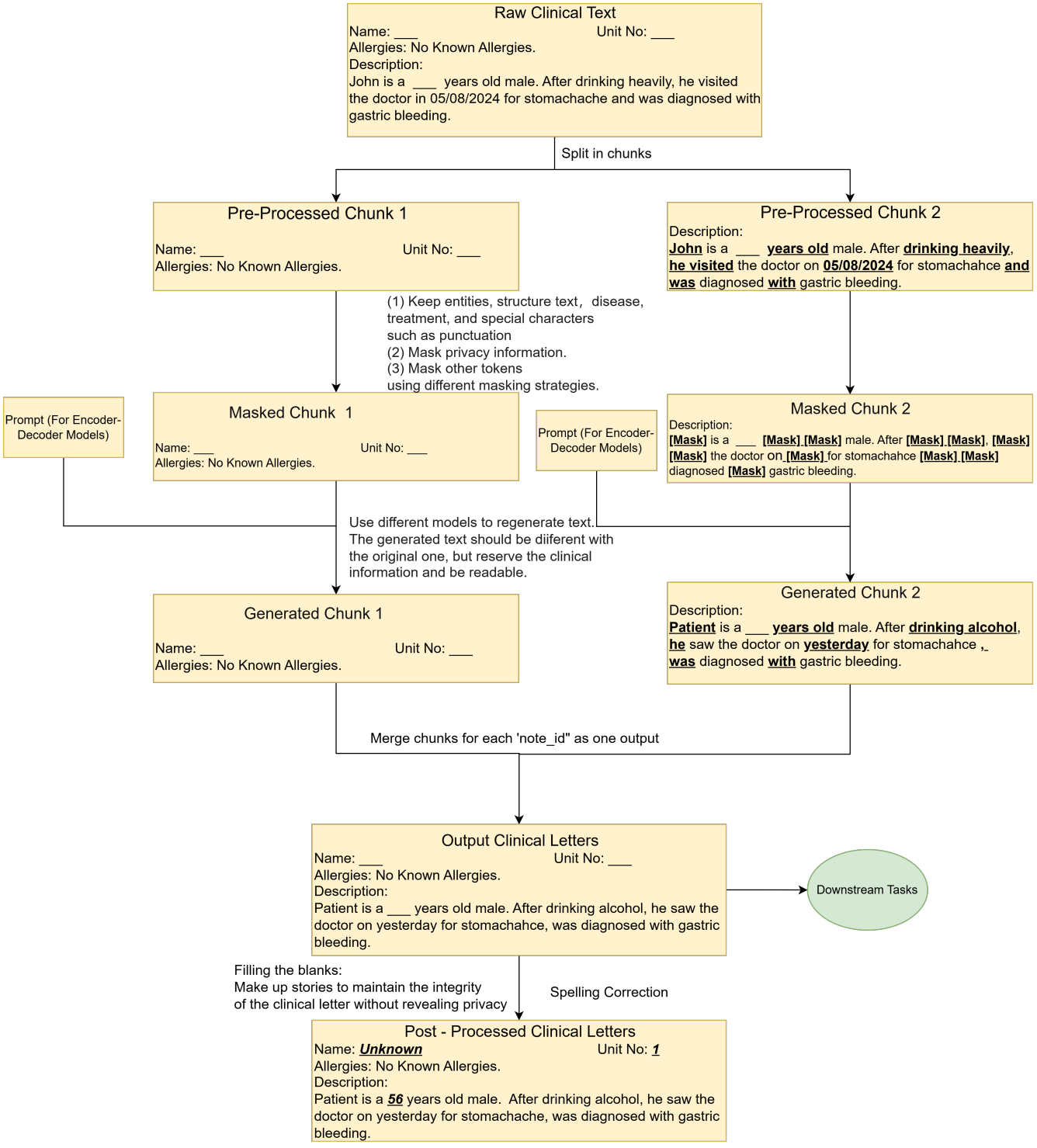

Figure 18. An Example of Masking and Generating

for masking. Although the initial intention was to mask 50% of tokens, the actual masking ratio was lower due to the requirement to preserve certain entities and structures.

max_lines	10	20	30	35	40	41	42	45	50
Inference Time (min)	13:47	8:10	6:44	5:24	5:10	5:01	5:12	5:54	6:05
Average Tokens Per Chunk	51.59	90.23	131.26	136.55	144.34	146.43	146.43	146.43	146.43

Table 6. Relation between Chunk Sizes and (Model Inference Time, Average Token Number)

```
History of Present Illness:
Chief Complaint:  ankle pain
Reason for Orthopedics Consult: management of open fracture

HISTORY OF PRESENT ILLNESS:
Patient is a ___ yo male previously healhty presenting w/ fall
from 6 feet, from ladder.
```

Figure 19. Original Unprocessed Example Sentence (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '10807423-DS-19') (The circled tokens will be masked)

Entity	Start	End	Concept ID
ankle pain	411	421	247373008
open fracture	468	481	397181002
fall	571	575	161898004

Table 7. Annotated Entities Extracted from the Example Sentence (They should be preserved from masking)

4.2.1 Encoder-Only Models

The original sentence is displayed in Fig 19. After the feature extraction, the resulting structure is shown in Fig 21. As detailed in Table 7, certain manually annotated entities are excluded from masking. The output of this masking process can be seen in Fig 20.

```
History of Present Illness:
Chief Complaint:  ankle pain
Reason for Orthopedics Consult: [MASK] [MASK] open fracture

HISTORY OF PRESENT ILLNESS:
[MASK] is a ___ yo male previously healhty [MASK] w/ fall
from 6 [MASK], from [MASK].
```

Figure 20. An Example of the Masked Sentence

The generated text using Bio_ClinicalBERT is displayed in Fig 22. For 'management of open fracture,' the model produced 'r', which is commonly used to denote 'right' in clinical contexts, showing a relevant and logical prediction. Furthermore, the model's input 'R ankle', despite not being in the figure due to

```
History of Present Illness :
Chief Complaint :
Reason for Orthopedics Consult :

HISTORY OF PRESENT ILLNESS :
```

Figure 21. Extracted Structure from the Example Sentence

```
History of Present Illness:
Chief Complaint:  ankle pain
Reason for Orthopedics Consult: r for open fracture

HISTORY OF PRESENT ILLNESS:
This is a ___ yo male previously healhty admitted w/ fall
from 6 stairs, from home.
```

Figure 22. Example Sentence Generated by Bio_ClinicalBERT

space constraints, provided context for predicting 'r' instead of 'left.' Interestingly, the term 'admitted' was generated even though it was not in the input, indicating the model's understanding of clinical context. Although the phrase 'from 6 stairs, from home' is entirely different from the original ('from 6 feet, from ladder'), it remains contextually appropriate.

Overall, Bio_ClinicalBERT produced a clinically sound sentence, even though no tokens matched the original. In other examples, the predicted words may partially overlap with the original text. Nonetheless, this model effectively retains clinical information and introduces diversity without altering the text's meaning.

The results from medicalai/ClinicalBERT and Clinical-Longformer are shown in Fig 23 and Fig 24. All three clinical-related models correctly predicted 'r' from the input context. Medicalai/ClinicalBERT performs *comparably* to Bio_ClinicalBERT, despite adding an extra comma, which did not affect the text's clarity. However, Clinical-Longformer's predictions, while understandable, were *repetitive* and less satisfactory. Importantly, none of these three models altered the original meaning.

```
History of Present Illness:
Chief Complaint:  ankle pain
Reason for Orthopedics Consult: r for open fracture

HISTORY OF PRESENT ILLNESS:
this is a ___ yo male previously healhty , w/ fall
from 6 feet, from home.
```

Figure 23. Example Sentence Generated by medicalai / ClinicalBERT

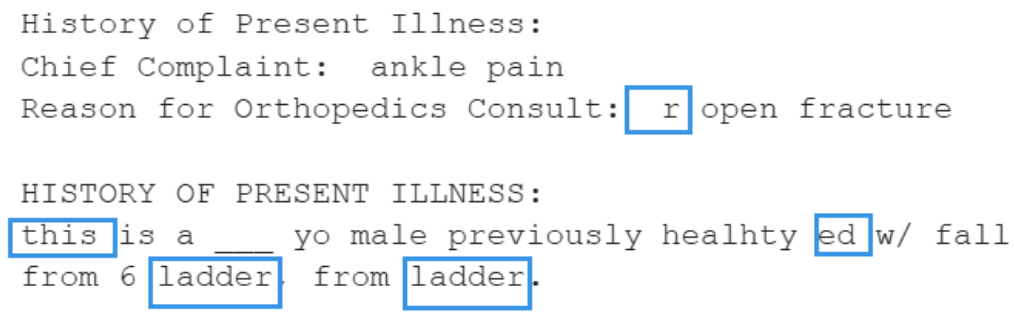

Figure 24. Example Sentence Generated by Clinical-Longformer

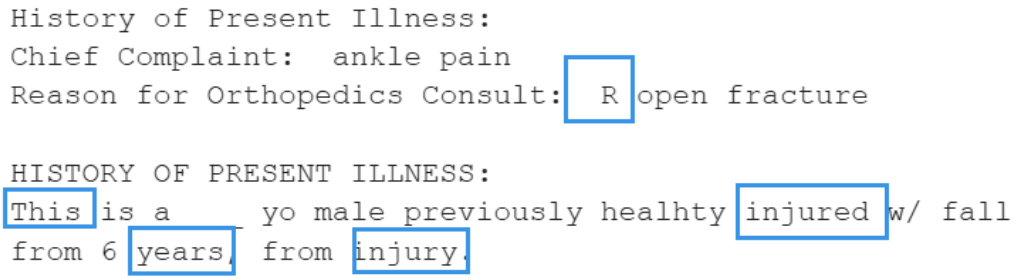

Figure 25. Example Sentence Generated by RoBERTa-base

The result generated by RoBERTa-base is shown in Fig 25. While the generated text initially seems reasonable, the predicted word 'years' shifts the focus to a temporal context, which was not intended. This is likely because RoBERTa is pre-trained on a general corpus and lacks sufficient clinical knowledge for accurate text generation, or it could simply be a coincidence based on this specific sentence, where RoBERTa-base inferred 'years' from its training data.

4.2.2 Decoder-Only GPT-4o

Additionally, **GPT-4o** was used for comparison, with the prompt "Replace '<mask>' with words in the following sentence: ". The results, shown in Fig 26, are satisfactory. As discussed in Section 2.3, decoder-only models excel in few-shot learning (Wu, 2024), which is confirmed by this experiment. However, its performance may decline with long clinical letters (Amin-Nejad et al., 2020b).

4.2.3 Encoder-Decoder Models

To further evaluate different PLMs in generating synthetic letters, we tested the T5 Family models. The generated results for the same sentence are shown in Fig 27, Fig 28, Fig 29, and Fig 30.

T5-base performs the best among these tested models. However, the results are still **not fully rational**, as it generated 'open is a ___ yo male'. The other three models tend to use de-identification (**DEID**) tags to replace the masked words, as these tags are part of their corpora. Furthermore, the T5 family models may predict multiple words for each token, aligning with findings in Section 2.3

All these four T5 family models perform **worse than the encoder-only** models. This is consistent with the findings from (Micheletti et al., 2024) that Masked Language Modelling (MLM) models outperform Causal Language Modeling (CLM) models in medical datasets.

Certainly! Here is the sentence with `<mask>` replaced by words:

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: severe open fracture

HISTORY OF PRESENT ILLNESS:
Patient is a ___ yo male previously healthy individual with fall from 6 feet
from a height.

Figure 26. Example Sentence Generated by GPT-4o

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: fracture dislocation, open fracture

HISTORY OF PRESENT ILLNESS:
open is a ___ yo male previously healhty fracture w/ fall
from 6 Reason from for

Figure 27. Example Sentence Generated by T5-base

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: on[DR][#] R open fracture

HISTORY OF PRESENT ILLNESS:
ankle is a ___ yo male previously healhty fracture w/ fall
from 6 dislocation[Name] from open

Figure 28. Example Sentence Generated by Clinical-T5-Base

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: , : open fracture

HISTORY OF PRESENT ILLNESS:
[name] is a ___ yo male previously healhty s w/ fall
from 6 r from or

Figure 29. Example Sentence Generated by Clinical-T5-Scratch

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: mechanical fall[MO][LOC][CI] open fracture

HISTORY OF PRESENT ILLNESS:
stairs is a ___ yo male previously healhty on w/ fall
from 6 6-13[URL] from fracture

Figure 30. Example Sentence Generated by Clinical-T5-Sci

4.3 Random Masking: Quantitative Results

4.3.1 Sentence-Level Quantitative Results: Encoder-Only Models

We first calculated representative quantitative metrics at the sentence level, matching the sample sentence used in Subsection 4.2. This approach allows for a better integration of quantitative and qualitative evaluations. Although SMOG is typically suited for medical datasets, it is less appropriate for sentence-level analysis, so the Flesch Reading Ease was used here. The results are shown in table 8.

	Model Evaluation			
	RoBERTa-base	medicalai / ClinicalBERT	Clinical-Longformer	Bio _ Clinical-BERT
ROUGE-1				
Generation Performance	86.54	88.46	89.52	84.91
Baseline	84.91	84.91	84.91	84.91
ROUGE-2				
Generation Performance	74.51	78.43	79.61	73.08
Baseline	73.08	73.08	73.08	73.08
ROUGE-L				
Generation Performance	86.54	88.46	89.52	84.91
Baseline	84.91	84.91	84.91	84.91
BERTScore F1				
Generation Performance	0.81	0.83	0.84	0.85
Baseline	0.79	0.65	0.79	0.65
METEOR				
Generation Performance	0.87	0.88	0.90	0.86
Baseline	0.85	0.85	0.85	0.85
Flesch Reading Ease				
Generation Performance	10.24	18.70	9.22	16.67
Baseline (Original)	8.21	8.21	8.21	8.21
Baseline (Mask)	16.67	16.67	16.67	16.67

Table 8. Encoder-Only Models Comparison at the Sentence Level (The 'Baseline' without annotations was calculated by comparing masked text to the original text)

Our objective is to generate letters that differ from the original while maintaining clinical semantics and structure. Thus, high ROUGE scores are not desired, as they indicate substantial **word/string** overlap. BERTScore is particularly useful for assessing **semantic** similarity, while METEOR offers a comprehensive evaluation considering word forms and synonyms theoretically. Flesch Reading Ease, on the other hand, provides a direct measure of textual **readability**.

We observed that clinical-related encoder-only models generally outperform RoBERTa-base in qualitative evaluation (see Subsection 4.2). However, from the quantitative perspective, RoBERTa-base shows mediocre performance across most metrics except for BERTScore. In contrast, Bio_ClinicalBERT, despite no word overlap in this sample sentence, achieves a reasonable clinical context and the highest BERTScore among the models. Both Medicalai/Clinical BERT and Bio_ClinicalBERT excel in Flesch Reading Ease, likely because they tend to predict tokens with fewer syllables words that preserve the original meaning.

Surprisingly, while METEOR is designed to closely reflect human evaluation, BERTScore appears to be more consistent with our evaluation criteria. This trend was observed in other sample texts as well. *Synthetic texts with higher BERTScore and lower ROUGE scores are more aligned with our objectives*. It is likely because BERTScore is calculated using word embeddings, which can capture deep semantic similarity more effectively. All evaluation results *meet or exceed the baseline, affirming the effectiveness of these four encoder-only models* in generating clinical letters.

4.3.2 Sentence-Level Quantitative Results: Encoder-Decoder Models

The evaluations for the encoder-decoder models, as shown in Table 9, generally underperform on most metrics compared to encoder-only models, except for METEOR. Interestingly, while the Flesch Reading Ease scores suggest a minimal impact on readability, the BERTScores are at least 0.05 lower than the baseline, indicating major deviations from the original meaning. This is consistent with our qualitative observations that the outputs from encoder-decoder models are largely unintelligible.

	Model Evaluation			
	T5-base	Clinical-T5-base	Clinical-T5-Scratch	Clinical-T5-Sci
ROUGE-1				
Generation Performance	86.79	85.19	87.38	80.36
Baseline	73.77	73.77	73.77	73.77
ROUGE-2				
Generation Performance	75.00	71.70	75.25	69.09
Baseline	63.33	63.33	63.33	63.33
ROUGE-L				
Generation Performance	84.91	83.33	87.38	80.36
Baseline	73.77	73.77	73.77	73.77
BERTScore F1				
Generation Performance	0.44	0.40	0.45	0.40
Baseline	0.50	0.50	0.50	0.50
METEOR				
Generation Performance	0.85	0.83	0.83	0.82
Baseline	0.85	0.85	0.85	0.85
Flesch Reading Ease				
Generation Performance	8.21	8.21	19.71	8.21
Baseline (Original)	8.21	8.21	8.21	8.21
Baseline (Mask)	8.21	8.21	8.21	8.21

Table 9. Encoder-Decoder Models Comparison at the Sentence Level (The Baseline without annotations was calculated by comparing masked text to the original text)

Collectively, the quantitative and qualitative results demonstrate that ***encoder-decoder models are not well-suited for generating clinical letters***, as they fail to preserve the original narratives. These results also

support the validity of using BERTScore as the primary evaluation metric, with other metrics serving as supplementary references. We also tested this on the **entire dataset**, which produced *consistent* results.

4.3.3 Quantitative Results on the Full Dataset: Encoder-Only Models

Based on the findings above, we expect a higher BERTScore and a lower ROUGE Score. We used the 0.4 masking ratio to illustrate the model comparison on the full dataset in Table 10. The other masking ratios show similar trends. Surprisingly, all encoder-only models this time showed comparable results, which contradicts our hypothesis that 'Clinical-related' models would outperform base models. This suggests that *training on the clinical dataset has limited impact on the quality of synthetic letters*. This may be because most clinical-related tokens are preserved, with only the remaining tokens being eligible for masking. Consequently, the normal encoder-only models can effectively understand the context and predict appropriate words while preserving clinical information. This differs slightly from the sentence-level comparisons, likely because the evaluation of a single sentence cannot fully represent the overall results. Despite this, BERTScore as a primary evaluation metric remains useful, as the correspondence between qualitative and quantitative evaluation is consistent, whether at the sentence or dataset level.

	Model Evaluation			
	RoBERTa-base	medicalai / ClinicalBERT	Clinical-Longformer	Bio_ Clinical-BERT
ROUGE-1				
Generation Performance	92.98	93.63	94.66	93.18
Baseline	85.64	85.44	85.64	85.61
ROUGE-2				
Generation Performance	86.10	87.42	89.50	86.50
Baseline	74.96	74.64	74.96	74.92
ROUGE-L				
Generation Performance	92.54	93.22	94.38	92.71
Baseline	85.64	85.44	85.64	85.61
BERTScore F1				
Generation Performance	0.91	0.90	0.92	0.90
Baseline	0.82	0.63	0.82	0.63

Table 10. Encoder-Only Models Comparison on the Full Dataset with Masking Ratio 0.4 (The Baseline was calculated by comparing masked text to the original text)

We will now explore how different *masking ratios* affect the quality of synthetic clinical letters. For each model, we generated data with masking ratios from 0.0 to 1.0, in increments of 0.1 (the masking ratios here refer only to the eligible tokens, as described in Subsection 3.4.3, and do not represent the actual overall masking ratio). Due to space limitations, we will present only the results for Bio_ClinicalBERT with a 0.2 increment here.

Table 11 shows that the higher masking ratio, the lower the similarity (metrics' scores) will be. As we expected, all evaluation values are higher than the baseline, but still below '1'. This means the model can understand the clinical context and generate understandable text. It is surprising that with the masking ratio of 1.0, BERTScore increased from the baseline (0.29) to 0.63. Although this score is not very high, it still reflects that Bio_ClinicalBERT can generate clinical text effectively.

In Table 12, we calculated three **readability** metrics, which were mentioned in Section 3.5. None of these metrics showed significant differences from the original ones. However, it is strange that the SMOG and Flesh-Kincaid Grade are not always between the original baseline and mask baseline. When the masking

Bio_ClinicalBERT	**Masking Ratio**					
	1.0	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
ROUGE-1						
Generation Performance	76.28	83.75	88.91	93.18	96.76	99.51
Baseline	64.05	71.56	78.56	85.61	92.63	99.22
ROUGE-2						
Generation Performance	62.60	70.77	78.81	86.50	93.42	99.02
Baseline	51.72	57.88	65.38	74.92	86.27	98.61
ROUGE-L						
Generation Performance	74.33	81.69	87.71	92.71	96.65	99.50
Baseline	64.05	71.56	78.56	85.61	92.63	99.22
BERTScore						
Generation Performance	0.63	0.75	0.83	0.90	0.95	0.99
Baseline	0.29	0.39	0.50	0.63	0.79	0.98
METEOR						
Generation Performance	0.70	0.80	0.87	0.93	0.97	1.00
Baseline	0.66	0.72	0.78	0.85	0.92	0.99

Table 11. Standard NLG Metrics Across Different Masking Ratios Using Bio_ClinicalBERT (The Baseline was calculated by comparing masked text to the original text)

ratio is high, the evaluation values even fall below both the masking and the original baseline. This may be because a *higher masking ratio leads to a lower valid prediction rate*. If the predicted words include many spaces or punctuation marks, the readability will decrease obviously.

Bio_ClinicalBERT	**Masking Ratio**					
	1.0	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
SMOG						
Generation Performance	8.91	9.18	9.50	9.79	10.00	10.13
Baseline (Original)	10.16	10.15	10.15	10.15	10.15	10.15
Baseline (Mask)	9.04	9.29	9.52	9.74	9.95	10.13
Flesch Reading Ease						
Generation Performance	63.77	63.44	61.41	59.54	58.06	57.02
Baseline (Original)	56.85	56.87	56.87	56.87	56.87	56.87
Baseline (Mask)	70.11	67.39	64.75	62.15	59.62	57.13
Flesch-Kincaid Grade						
Generation Performance	7.32	7.70	8.24	8.66	9.01	9.22
Baseline (Original)	9.26	9.26	9.26	9.26	9.26	9.26
Baseline (Mask)	7.41	7.79	8.16	8.52	8.87	9.22

Table 12. Readability Metrics Across Different Masking Ratios Using Bio_ClinicalBERT (The Baseline without annotations was calculated by comparing masked text to the original text)

In Table 13, considering the perplexity, the masking baseline is very high, while the values for synthetic letters are close to the original ones. This indicates that the synthetic letters are useful for training clinical models. For information entropy, regardless of the masking ratio, it can *effectively preserve the amount of information*. As for subjectivity, since all the values are close, we don't need to worry that the synthetic letters will be biased.

As shown in Table 14, **inference time** for the entire dataset consistently ranges between 3 to 4 hours. However, it decreases with either very high or very low masking ratios. A mid-range masking ratio of approximately 0.6 results in longer inference times, likely because lower ratios reduce the number of

Bio_ClinicalBERT	Masking Ratio					
	1.0	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
Perplexity						
Generation Performance	2.24	2.32	2.31	2.30	2.29	2.29
Baseline (Original)	2.22	2.28	2.28	2.28	2.28	2.28
Baseline (Mask)	250.37	65.42	24.29	8.95	4.03	2.39
Information Entropy						
Generation Performance	5.46	5.80	5.92	5.96	5.98	5.98
Baseline (Original)	5.98	5.98	5.98	5.98	5.98	5.98
Baseline (Mask)	4.51	4.93	5.29	5.60	5.85	5.97
Subjectivity						
Generation Performance	0.32	0.32	0.32	0.32	0.33	0.33
Baseline (Original)	0.33	0.33	0.33	0.33	0.33	0.33
Baseline (Mask)	0.41	0.39	0.38	0.37	0.35	0.33

Table 13. Advanced Text Quality Metrics Across Different Masking Ratios Using Bio_ClinicalBERT (The Baseline without annotations was calculated by comparing masked text to the original text)

masked tokens to process, while higher ratios provide less context, reducing the computational load. This lack of effective context also increases the invalid prediction rate. Conversely, with a masking ratio of '0', even a small number of prediction errors can substantially affect the overall accuracy, as only a few tokens are masked.

Masking Ratio	**1.0**	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
Inference Time	3:12:05	3:28:56	3:33:26	3:25:16	3:13:26	3:01:11
Invalid Prediction Rate	0.72	0.47	0.34	0.28	0.25	0.37

Table 14. Inference Time and Invalid Prediction Rate Across Different Masking Ratios Using Bio_ClinicalBERT

4.4 Other Masking Strategies Using Bio_ClinicalBERT

Masked Sentence

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: [MASK] of open fracture

HISTORY OF PRESENT ILLNESS:
[MASK] [MASK] [MASK] ___ yo male previously healhty presenting w/ fall from 6 [MASK], [MASK] ladder.

Synthetic Sentence

History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: repair of open fracture

HISTORY OF PRESENT ILLNESS:
Chief ##t _ ___ yo male previously healhty presenting w/ fall from 6 stairs, hit ladder.

Figure 31. Example Sentence 1 with Different Masked Tokens

Masked Sentence
History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: [MASK] [MASK] open fracture

HISTORY OF PRESENT ILLNESS:
Patient is a ___ [MASK] [MASK] previously [MASK] [MASK] w/ fall
from 6 feet, from ladder.

Synthetic Sentence
History of Present Illness:
Chief Complaint: ankle pain
Reason for Orthopedics Consult: r R open fracture

HISTORY OF PRESENT ILLNESS:
Patient is a ___ _ , previously healthy male w/ fall
from 6 feet, from ladder.

Figure 32. Example Sentence 2 with Different Masked Tokens

There is a random selection when masking tokens at certain ratios. Masking different types of tokens will lead to different results, as shown in Fig 31 and Fig 32. This variability is understandable since the encoder-only models use bidirectional attention, as mentioned in Section 2.3. *These models need to predict the masked tokens based on the context*. Therefore, it is necessary to experiment with different masking strategies based on the types of tokens. We used **POS** tagging and **stopwords** to observe how these strategies influence the quality of synthetic letters.

As discussed in Subsection 4.3, BERTScore should be the primary evaluation metric for our objective. Additionally, the invalid prediction rate is useful for assessing the model's ability to generate informative predictions, and ROUGE scores help evaluate literal diversity. Therefore, these quantitative metrics, calculated using different masking strategies, will be shown in this section. Similar to Subsection 4.3, we experimented with different masking ratios calculated from the eligible tokens (masked tokens divided by eligible tokens). The ratios are increased in increments of 0.1, ranging from 0.0 to 1.0. Due to space constraints, only metrics with increments of 0.2 will be shown here. A comparison with the same actual masking ratio (masked tokens divided by total tokens in the text) will also be presented in this subsection.

4.4.1 Masking Only Nouns

Nouns often correspond to Personally Identifiable Information (**PII**), so masking nouns can serve as a verification step for de-identification.

As shown in Table 15, **the fewer nouns we mask, the better all these metrics perform**. This trend is consistent with random masking. When the noun masking ratio is 1.0, meaning all nouns are masked, BERTScore increases from a baseline of 0.70 to 0.89. This means the *model predicted meaningful nouns*. A similar trend is observed in the ROUGE scores. All evaluations are higher than the baseline but lower than '1'. However, ROUGE scores show a smaller improvement than BERTScore. This may be because the model generates synonyms or paraphrases that retain the original meaning. As the nouns masking ratio increases from 0.0 to 1.0, the BERTScore drops from 0.99 to 0.89, indicating a significant decrease.

Bio_ClinicalBERT	**Nouns Masking Ratio**					
	1.0	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
ROUGE-1						
Generation Performance	93.29	95.16	96.48	97.62	98.66	99.51
Baseline	88.13	90.79	92.98	95.19	97.39	99.22
ROUGE-2						
Generation Performance	86.71	90.29	92.84	95.12	97.28	99.02
Baseline	78.32	82.92	86.79	90.82	95.01	98.61
ROUGE-L						
Generation Performance	93.00	94.96	96.35	97.56	98.64	99.50
Baseline	88.13	90.79	92.98	95.19	97.39	99.22
BERTScore						
Generation Performance	0.89	0.92	0.94	0.96	0.98	0.99
Baseline	0.70	0.76	0.81	0.86	0.92	0.98
Invalid Prediction Rate						
Generation Performance	0.37	0.34	0.33	0.32	0.32	0.37

Table 15. Quantitative Comparisons of Nouns Masking Ratios (The 'Baseline' was calculated by comparing masked text to the original text)

Therefore, to generate synthetic clinical letters that are distinguishable but still retain the original clinical information, we can only partially mask nouns (around a 0.8 masking ratio). It helps maintain balanced evaluation scores. When all nouns are masked, the quality of synthetic letters deteriorates, with the BERTScore falling below 0.9 and the invalid prediction rate rising to 0.37.

4.4.2 Masking Only Verbs

Masking only verbs also helps identify which token types are appropriate for masking to achieve our objective. While verbs are essential to describing clinical events, some can still be inferred from context. Therefore, *masking verbs may have a slight effect on the quality of synthetic clinical letters, but it can also introduce some variation*.

Table 16 shows a similar trend for masking verbs as observed with other masking strategies in standard NLG metrics. However, it is surprising that as the masking ratio increases, both the invalid prediction rate and NLG metrics decrease. This phenomenon can be attributed to two main reasons. **First**, the model seems to prioritise predicting meaningful tokens (rather than punctuation, spaces, etc.) to generate coherent sentences. Contextual relevance is only considered after the sentence structure is complete. This may be due to the important role of verbs in sentences. **Second**, the original raw data may contain fewer verbs than nouns. Therefore, the number of actual masked tokens changes slightly when verbs are masked, making the model less sensitive to them. This is also reflected in BERTScore. If all verbs are masked, the BERTScore remains high at 0.95, whereas if all nouns are masked, the BERTScore drops to 0.89.

4.4.3 Masking Only Stopwords

As mentioned in Subsection 3.4.3, masking stopwords aims to reduce noise for model understanding while introducing variation in synthetic clinical letters. Table 17 shows that *masking only stopwords follows a similar trend to random masking*, where a higher masking ratio leads to lower ROUGE Score and BERTScore. Additionally, the Invalid Prediction Rate is at its lowest with a medium masking ratio. This is because higher masking ratios always result in more information loss. On the other hand, lower masking ratios lead to fewer tokens being masked, which makes small prediction errors more influential. The results show an overall low Invalid Prediction Rate and high BERTScore, indicating that *stopwords have only*

Bio_ClinicalBERT	Verb Masking Ratio					
	1.0	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
ROUGE-1						
Generation Performance	96.48	97.38	97.97	98.54	99.08	99.51
Baseline	94.11	95.50	96.48	97.50	98.48	99.22
ROUGE-2						
Generation Performance	92.79	94.63	95.84	97.04	98.15	99.02
Baseline	88.53	91.26	93.18	95.19	97.14	98.61
ROUGE-L						
Generation Performance	96.37	97.31	97.92	98.51	99.07	99.50
Baseline	94.11	96.48	96.48	97.50	98.48	99.22
BERTScore						
Generation Performance	0.95	0.97	0.97	0.98	0.99	0.99
Baseline	0.82	0.86	0.89	0.92	0.95	0.98
Invalid Prediction Rate						
Generation Performance	0.31	0.31	0.32	0.32	0.33	0.37

Table 16. Quantitative Comparisons of Verb Masking Ratios (The 'Baseline' was calculated by comparing masked text to the original text)

a limited influence on the model's understanding of context. This is not because the original raw letters contain very few stopwords. In fact, there are even more stopwords than nouns and verbs, as seen in sample texts.

Bio_ClinicalBERT	Stopwords Masking Ratio					
	1.0	**0.8**	**0.6**	**0.4**	**0.2**	**0.0**
ROUGE-1						
Generation Performance	92.84	95.17	96.56	97.71	98.69	99.51
Baseline	81.52	85.53	89.04	92.54	96.04	99.22
ROUGE-2						
Generation Performance	84.64	89.41	92.53	95.05	97.24	99.02
Baseline	68.30	74.35	79.99	86.02	92.44	98.61
ROUGE-L						
Generation Performance	91.84	94.53	96.23	97.56	98.65	99.50
Baseline	81.52	85.53	89.04	92.54	96.04	99.22
BERTScore						
Generation Performance	0.89	0.93	0.95	0.97	0.98	0.99
Baseline	0.57	0.65	0.71	0.79	0.88	0.98
Invalid Prediction Rate						
Generation Performance	0.29	0.22	0.20	0.18	0.19	0.37

Table 17. Quantitative Comparisons of Stopwords Masking Ratios (The 'Baseline' was calculated by comparing masked text to the original text)

4.4.4 Comparison of Identical Actual Masking Ratios

To further observe how different masking strategies influence the generation of clinical letters, we compared the results using the same actual masking ratios but with different strategies. In other words, the number of masked tokens is fixed, so the only variable is *the type of tokens being masked*. Table 18 shows the results with a 0.04 actual masking ratio, and Table 19 shows the results with a 0.1 actual masking ratio.

As we can see, masking only stopwords got the highest BERTScore and lowest invalid prediction rate. Therefore, stopwords have little influence on the overall meaning of the text, which is consistent with our

earlier findings. Additionally, masking nouns and verbs perform worse than random masking. Therefore, if we want to preserve the original meaning, we cannot mask too many nouns and verbs.

Bio_ ClinicalBERT	**Masking Strategies**			
	Noun Masking (0.4)	**Stopwords Masking (0.2)**	**Verbs Masking (0.8)**	**Randomly Masking (0.1)**
ROUGE-1				
Generation Performance	97.62	98.69	97.62	98.28
Baseline	95.19	96.04	95.19	96.16
ROUGE-2				
Generation Performance	95.12	97.24	95.12	96.50
Baseline	90.82	92.44	90.82	92.68
ROUGE-L				
Generation Performance	97.56	98.65	97.56	98.25
Baseline	95.19	96.04	95.19	96.16
BERTScore				
Generation Performance	0.96	0.98	0.96	0.97
Baseline	0.86	0.88	0.86	0.88
Invalid Prediction Rate				
Generation Performance	0.32	0.19	0.32	0.25

Table 18. Quantitative Comparison of Different Masking Strategies at a 0.04 Actual Masking Ratio (The 'Baseline' was calculated by comparing masked text to the original text)

Bio_ClinicalBERT	**Nouns Masking (1.0)**	**Stopwords Masking (0.6)**	**Random Masking (0.3)**
ROUGE-1			
Generation Performance	93.29	96.56	95.10
Baseline	88.13	89.04	89.16
ROUGE-2			
Generation Performance	86.71	92.53	90.17
Baseline	78.32	79.99	80.44
ROUGE-L			
Generation Performance	93.00	96.23	94.86
Baseline	88.13	89.04	89.16
BERTScore			
Generation Performance	0.89	0.95	0.93
Baseline	0.70	0.71	0.71
Invalid Prediction Rate			
Generation Performance	0.37	0.20	0.26

Table 19. Quantitative Comparisons of 0.1 Actual Masking Ratio (The Baseline was calculated by comparing masked text to the original text)

4.4.5 Hybrid Masking

After comparing different strategies with the same actual masking ratio, we explored hybrid masking strategies and compared them with other strategies at the same actual ratio. The results are shown in Table 20. The first three columns have the same actual masking ratio. Masking only stopwords achieved the strongest performance among these strategies. However, when nouns are also masked along with stopwords,

performance decreases, as masking nouns negatively affects the results. Despite this, it still performs better than random masking, indicating that stopwords have a greater influence than nouns. Next, we compared the last two columns. If 0.5 of nouns and 0.5 of stopwords are masked, adding an additional 0.5 of masked verbs leads to worse performance, showing that verbs also negatively influence the model's performance.

Bio_ClinicalBERT	**Stopwords Masking (0.8)**	**Random Masking (0.4)**	**Nouns (0.5) and Stopwords (0.5)**	**Nouns (0.5), Verbs (0.5), Stopwords (0.8)**
Actual Masking Ratio	0.13	0.13	0.13	0.16
ROUGE-1				
Generation Performance	95.17	93.18	94.29	91.34
Baseline	85.53	85.61	85.98	82.47
ROUGE-2				
Generation Performance	89.41	86.50	88.34	83.08
Baseline	74.35	74.92	75.30	70.73
ROUGE-L				
Generation Performance	94.53	92.71	93.80	90.50
Baseline	85.53	85.61	85.98	82.47
BERTScore				
Generation Performance	0.93	0.90	0.91	0.87
Baseline	0.65	0.63	0.65	0.57
Invalid Prediction Rate				
Generation Performance	0.22	0.28	0.28	0.31

Table 20. Quantitative Comparisons for Hybrid Masking (The Baseline was calculated by comparing masked text to the original text)

4.4.6 Comparison with and without (w/o) Entity Preservation

To further explore whether keeping entities is useful for our task, we compared our results with a baseline that does not retain any entities. The baseline was trained with four epochs of fine-tuning on our dataset. Specifically, 0.4 of nouns from all tokens were randomly masked during the baseline training. In contrast, in our experiments, only eligible tokens—excluding clinical information—were selected for masking. The comparisons are shown in Table 21.

Bio_ClinicalBERT	**With Entity Preservation** (0.4 Nouns Masking)	**With Entity Preservation** (0.3 Random Masking)	**Without Entity Preservation** (0.4 Nouns Masking)
ROUGE-1	97.62	95.10	97.31
ROUGE-2	95.12	90.17	94.46
ROUGE-L	97.56	94.86	93.71
BERTScore	0.96	0.93	0.91

Table 21. Comparison with and without Entity Preservation Using Bio_ClinicalBERT

As we can see, **when 0.4 nouns are masked while preserving entities, the models perform much better** than those without any entity preservation. Interestingly, when we randomly mask 0.3 while preserving entities, the model achieves lower ROUGE-1 and ROUGE-2 scores but higher ROUGE-L and BERTScores compared to models without entity preservation. This trend is consistent across different settings. It suggests that models preserving entities have less overlap with the original text, while they

can retain the original narrative better. Additionally, the higher ROUGE-L score suggests that the step of **preserving document structure is indeed effective**.

These results also confirm our initial hypothesis that, for our objective — generating clinical letters that can keep the original meaning while adding some variety — **retaining entities** is much more effective than just fine-tuning the model. Moreover, this approach can effectively preserve useful information while avoiding overfitting.

4.5 Downstream NER Task

To further evaluate whether synthetic letters have the potential to replace the original raw letters, particularly in the domains of clinical research and model training, a downstream NER task was implemented. Two **spaCy NER** models were trained separately on **original raw letters and synthetic letters**. Specifically, the synthetic letters were generated with 0.3 random masking while preserving entities.

The spaCy models trained on original and synthetic letters showed **similar evaluation scores**. They even achieved F1 scores comparable to ScispaCy's score of 0.843. Therefore, the unmasked context appears to have minimal influence on model understanding. Consequently, *our synthetic letters can be used in NER tasks to replace real-world clinical letters, thereby further protecting sensitive information*.

Metric	spaCy Trained on Original Letters	spaCy Trained on Synthetic Letters	Performance Delta (Δ)
F1 Score	0.855	0.853	-0.002
Precision	0.865	0.863	-0.002
Recall	0.846	0.843	-0.003

4.6 Clinical Evaluation

4.6.1 Clinical Semantic Preservation

As mentioned in Subsection 3.5.4, we use BioGPT with a random masking ratio of 0.3 to evaluate the integrity of clinical narrative preservation. As shown below, the mean similarity score reaches 0.98, which is slightly higher than the score obtained using the BERTScore metric. This may be because BioGPT evaluates semantic similarity from a clinical perspective. Additionally, such a high score suggests that the synthetic clinical letters can potentially serve as replacements for the original ones.

Metric	Max Score	Min Score	Mean Score	Std Deviation	Evaluation Set Size
Value	0.9996	0.9037	0.9896	0.0147	204

4.6.2 Expert-Simulated Evaluation of Clinical Quality

As mentioned in Subsection 3.5.4, we prompted GPT-3.5-Turbo to simulate a clinical expert and evaluate clinical soundness and narrative coherence. The masked letters (with text replaced by '`<mask>`') continued to serve as a baseline. The results are shown in the Tabel 22

Clinical Soundness: The average clinical soundness score of the generated letters (0.604) is slightly lower than that of the original letters(0.766). Surprisingly, it is even lower than the score of the masked letters (0.611). We further identified all cases where the generated letters scored lower than the masked ones in clinical soundness. These cases account for 14% (29 out of 204) of the processed letters. One possible explanation is that Bio_ClinicalBERT occasionally produces hallucinatory content, which may obscure or distort the original clinical semantics. However, in the majority of cases, the generated letters achieve

Metric	Avg.	Max.	Min.	Std.
Clinical Soundness				
Baseline (Original)	0.766	1.0	0.5	0.11
Baseline (Masked)	0.611	1.0	0.5	0.147
Generation Performance	0.604	1.0	0.5	0.145
Narrative Coherence				
Baseline (Original)	0.664	0.8	0.3	0.082
Baseline (Masked)	0.418	0.7	0.2	0.168
Generation Performance	0.460	0.7	0.2	0.177

Table 22. Expert-Simulated Evaluation Results

clinical soundness scores that are comparable to the masked letters and close to those of the original ones — demonstrating the overall potential of our synthetic letters to replace real ones.

Narrative Coherence: As expected, the narrative coherence score of the generated letters (0.460) is slightly lower than that of the original ones (0.664), but higher than that of the masked letters (0.418). These results further support the feasibility of using synthetic letters as substitutes for real clinical letters.

4.7 Post-Process Results

4.7.1 Filling in the Blanks

One example text without post-processing is shown in Fig 33. After filling in the blanks, the results with BERT-base and Bio_ClinicalBERT are shown in Fig 34 and Fig 35 separately. We can see that both models can partially achieve the goal of making the text more complete. However, neither of them created a coherent story to fill in these blanks. They just used general terms like "hospital" and "clinic". Perhaps other decoder-only models, more suitable for generating stories like GPT, could perform better and should be explored in the future.

```
He initially presented to ___, where he
received nitropaste. Vitals there were 94/22, HR 94, RR 18, 97%
on RA, and no pain. Cardiac enzymes there were CK 170 and
Troponin 0.02 at 12:30 ___.
Upon arrival to ___, his blood pressure was 104/52, HR 52, RR
18, temperature 96.2, and respiratory rate of 18. He was given
325 mg of aspirin and tolerated it well--of note there is a
possible allergy to aspirin noted in his admission intake form.
```

Figure 33. Example Sentence Before Post-Processing (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '16441224-DS-19')

4.7.2 Spelling Correction

Fig 36 shows that if the incorrect words are masked, the models may be able to correct the misspelled tokens by predicting them. However, the masking process is random. Additionally, sometimes the predicted words will be incorrect because some models tokenize the sentence into word-pieces. Therefore, a post-processing step is necessary for correcting spelling.

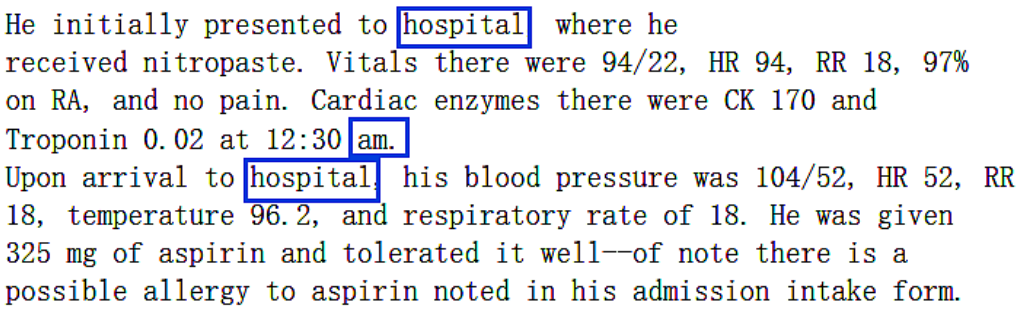

Figure 34. Post-Processing Results with BERT-Base

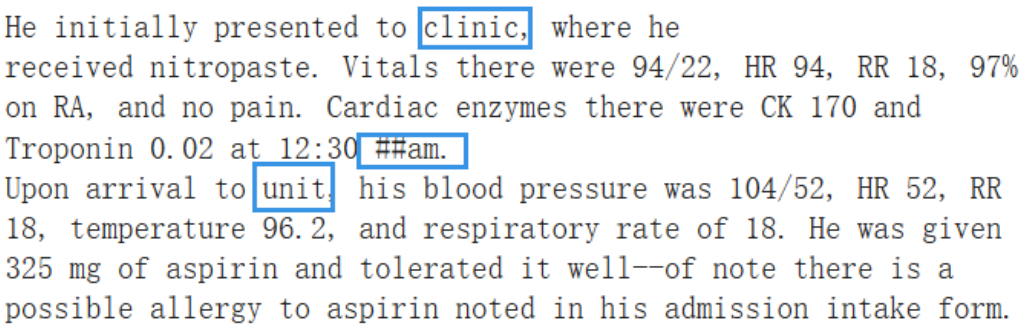

Figure 35. Post-Processing Results with Bio_ClinicalBERT

As shown in Fig 37, Tooltik 'TextBlob' (Loria, 2024) can successfully correct misspelled words ('healhty') in our sample text. However, if clinical entities are not preserved during the pre-processing step, 'TextBlob' (Loria, 2024) may misidentify some clinical terms as spelling errors. It may be because 'TextBlob' (Loria, 2024) was developed on the general corpus, not a clinical one. Additionally, its corrections are limited to the word level and do not consider any context. Therefore, if words are misspelled deliberately, they could be processed incorrectly. Thus, *developing a clinical misspelling correction toolkit is a promising* research direction in the future.

4.8 Discussion

We found that different masking strategies result in notable differences in model performance. To enhance the practical applicability of our research, we provide a guideline for selecting appropriate masking strategies for different scenarios, as shown in the Table 23.

As mentioned earlier, we observe that when most clinical terms are preserved, fine-tuning the model may not be necessary. In terms of clinical evaluation, hallucinated content was found to negatively affect clinical soundness, suggesting that retrieval-augmented generation (RAG) or integration with a clinical knowledge graph may be beneficial for future improvements. Further exploration is also needed—such as dynamic vocabulary construction — to better handle clinical abbreviations and novel terms. Our synthetic framework for clinical letters did not show any notable negative effects on narrative coherence or semantic preservation, and the high performance in downstream NER tasks further supports the feasibility of using synthetic letters as substitutes for original ones. Although filling in blanks and correcting spelling errors are essential for improving text quality, mitigating errors in processing rare clinical terms remains a major challenge, as previously discussed.

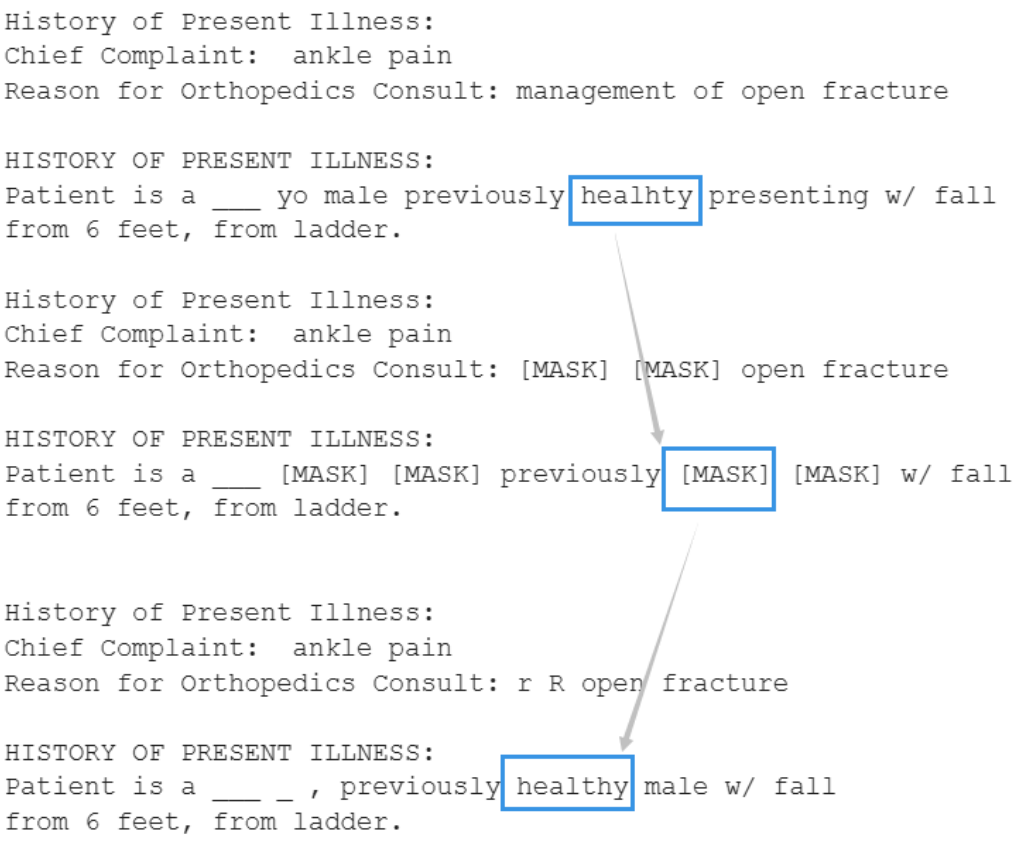

Figure 36. Spelling Correction by masking and Generating (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '10807423-DS-19')

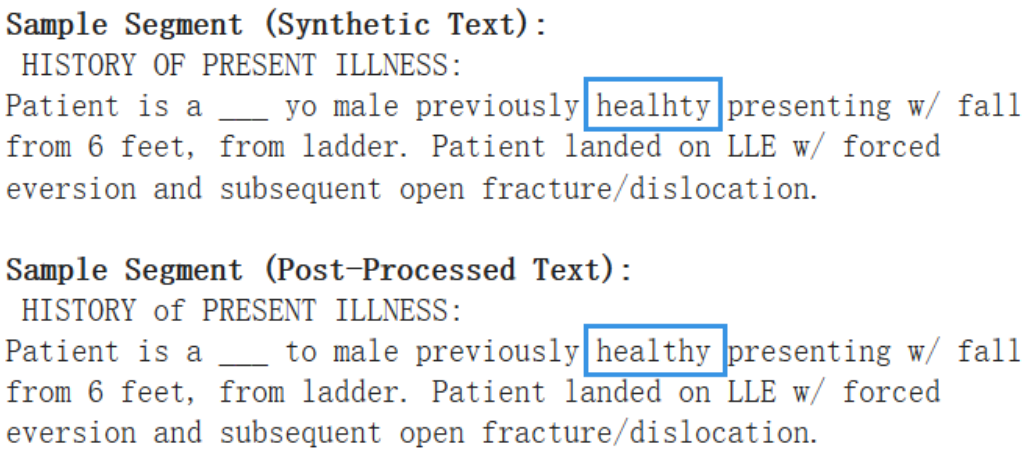

Figure 37. Spelling Correction (A et al., 2000; Johnson et al., 2024a, 2023b) ('note_id': '10807423-DS-19')

5 CONCLUSIONS AND FUTURE WORK

5.1 Key Findings

These results provided some useful findings in generating clinical letters, including:

Priority	Note	Suggested Masking Strategy	Application Scenarios
Diversity First	To improve the model's generalisation.	Random Masking (Primarily), Clinical Terms Masking(Limited).	Basic clinical model pre-training; Data augmentation
Clinical Soundness First	The synthetic letters should satisfy clinical factuality	Keep Clinical Terms (Complete); Stopwords Masking (Extensive); Verbs/Nouns Masking	Clinical education; Clinical QA model training; Clinical model fine-tuning
Privacy First	To prevent PHI disclosure and mitigate privacy reconstruction through adversarial attacks.	Private Tokens Masking (Complete); Nouns Masking (Extensive); Verbs/Stopwords Masking (Medium)	Building open-source datasets; Commercial deployment

Table 23. Priority-Based Masking Guidelines

- **Encoder-only models generally perform much better** in clinical-letter masking and generation tasks, which is consistent with a very recent study by (Micheletti et al., 2024). When clinical information is preserved, **base encoder-only models perform comparably to clinical-related models**.
- To generate clinical letters that preserve clinical narrative while adding variety, **BERTScore should be the primary evaluation metric**, with other metrics serving as supporting references. This is because BERTScore focuses more on semantic rather than literal similarity, and it is consistent with qualitative assessment results.
- **Different types of masked tokens influence** the quality of synthetic clinical letters. **Stopwords** have a **positive** impact, while **nouns and verbs** have **negative** impacts.
- For our objective, **preserving useful tokens** is more effective than just fine-tuning the model without preserving any entities.
- The **unmasked context** has minimal influence on the models' understanding. As a result, the **synthetic letters can be effectively used in downstream NER task** to replace original real-world letters.
- The synthetic letters **largely preserve** the **consistency** and **coherence** of clinical narratives from the original letters. However, Bio_ClinicalBERT occasionally generates **hallucinated** content, which may **negatively** impact clinical soundness and factuality.

5.2 Limitations

Although the strategies mentioned above help generate diverse, de-identified synthetic clinical letters, there are still some limitations in applying this method generally.

- **Challenges in the Dataset:** Since these clinical letters are derived from the real world, certain issues are inevitable. For example, there may be spelling errors in the dataset. In note_id "10807423-DS-19", the word 'healthy' is misspelled as 'healhty'. Such errors can negatively impact the usability of the synthetic text. Additionally, some polysemous words may cause contextual ambiguity. For instance, the word 'back' can refer to an anatomical entity (e.g., the back of the body), or be used as an adverb.
- **Data Volume:** Due to the difficulty in collecting annotated data, only 204 clinical letters are included in our research. This limited sample size may not be sufficiently representative, which could restrict the generalizability of our findings to a broader scenario. Moreover, the data we used were already de-identified. Although we considered de-identification and took steps to mask all private information,

the effectiveness of these approaches cannot be thoroughly evaluated, as we do not have access to sensitive datasets.

- **Evaluation Metrics:** In this paper, we primarily use BERTScore as our main evaluation metric, while also incorporating other metrics such as ROUGE and readability metrics. However, there is currently **no comprehensive evaluation framework that can assess all aspects** simultaneously, including maintaining the original meaning, diversity, readability, clinical soundness, and even privacy protection effectiveness.
- **Clinical Knowledge Understanding:** While the model can often preserve clinical entities and generate contextually reasonable tokens, it sometimes makes comprehension errors. For example, in a context where 'LLE' ('left lower extremity') is used, Bio_ClinicalBERT incorrectly predicts the nearby masked token as 'R ankle' ('right ankle'). In this case, the model fails to accurately capture the side clinical knowledge. Another challenge lies in handling long-tail phenomena and understanding abbreviated expressions, which are common in clinical language. Although spell correction techniques are explored in our project, distinguishing between a genuinely novel term and a simple misspelling remains difficult.
- **Computing Resources:** Due to resource limitations, we explored a limited range of language generation models. Alternative architectures — such as enhanced decoder-only models — may be more suitable for our task.

5.3 Future Work

Based on the limitations mentioned above, we outlined some potential directions to further explore:

- **Test on More Clinical Datasets:** To further evaluate the effectiveness of these masking strategies, more annotated clinical letters should be tested to assess system generalisation.
- **Assess De-identification Performance:** A quantitative metric for de-identification evaluation should be included in the future. Non-anonymous synthetic datasets can be used to evaluate the de-identification process, so that this system can be applied directly to real-world clinical letters in the future.
- **GRPO-based Reinforcement Learning:** The Group Relative Policy Optimization (GRPO) algorithm, as proposed in DeepSeek (Shao et al., 2024), has the potential to effectively balance multiple objectives, including clinical soundness, semantic integrity, textual diversity, and de-identification quality.
- **Evaluation Benchmark:** A new metric that is suitable for our task should be developed. Specifically, this metric should consider both similarity and diversity. Weighting parameters for each dimension could be useful and can be obtained through neural networks. For evaluating clinical soundness, it is necessary to invite more clinicians to assess the synthetic letters based on multiple dimensions (Ellershaw et al., 2024). Furthermore, the mapping from clinical letters to their quality scores can be learned using deep learning.
- **Balancing Knowledge from Both Clinical and General Domains:** Although there are numerous clinical-related encoder-only models, only a few can effectively integrate clinical and general knowledge. (Xie et al., 2024) demonstrated that mixing the clinical dataset with the general dataset in a certain proportion can help the model better understand clinical knowledge. Therefore, a new BERT-based model should be trained from scratch using both clinical and general domain datasets.
- **Synonymous Substitution:** We focused on exploring the range of eligible tokens for masking. Additionally, a masking strategy similar to BERT's can be integrated with our results (Devlin et al.,

2019). Specifically, we can select certain tokens to mask, some to retain, and replace others with synonyms. This approach can further enhance the variety of synthetic clinical letters. Moreover, the retained clinical entities can also be substituted using the entity linking to SNOMED CT.

- **Spelling Correction:** As mentioned in Section 4.7, there are very few toolkits available for spelling correction in the clinical domain. Standard spelling correction tools may misidentify clinical terms as misspelled words. Therefore, it is necessary to develop a specialised spell-checking tool adapted to the clinical domain.

ETHICS CONSIDERATION

There are no ethical concerns in this paper. Before accessing the dataset, we completed the necessary training (CITI Data or Specimens Only Research) and signed the Data Use Agreement (DUA). In addition, the results of this paper will not be used for any commercial purpose.

ACKNOWLEDGEMENTS

We thank the insightful feedback from Nicolo Micheletti which is very valuable to the start of this work. LH, WDP, and GN are grateful for the support from the grant "Assembling the Data Jigsaw: Powering Robust Research on the Causes, Determinants and Outcomes of MSK Disease." The project has been funded by the Nuffield Foundation, but the views expressed are those of the authors and not necessarily of the Foundation. Visit www.nuffieldfoundation.org. LH, WDP, and GN are also supported by the grant "Integrating hospital outpatient letters into the healthcare data space" (EP/V047949/1; funder: UKRI/EPSRC). LH is grateful to the EU project 4D Picture, funded by EU Horizon Europe Work Programme 101057332. Views are the authors' only.

REFERENCES

[Dataset] A, G., L, A., L, G., J, H., PC, I., R, M., et al. (2000). Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. `https://physionet.org/content/`

Abouelmehdi, K., Beni-Hessane, A., and Khaloufi, H. (2018). Big healthcare data: preserving security and privacy. *Journal of big data* 5, 1–18

Acheampong, F. A., Nunoo-Mensah, H., and Chen, W. (2021). Transformer models for text-based emotion detection: a review of bert-based approaches. *Artificial Intelligence Review* 54, 5789–5829

[Dataset] AI, M. (2023). Llama3 model card. `https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md`. Accessed on: 04/05/2024

Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., et al. (2019). Publicly available clinical bert embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*. 72–78

Amin-Nejad, A., Ive, J., and Velupillai, S. (2020a). Exploring transformer text generation for medical dataset augmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*. 4699–4708

Amin-Nejad, A., Ive, J., and Velupillai, S. (2020b). Exploring transformer text generation for medical dataset augmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, eds. N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara,

B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis (Marseille, France: European Language Resources Association), 4699–4708

Banerjee, S. and Lavie, A. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, eds. J. Goldstein, A. Lavie, C.-Y. Lin, and C. Voss (Ann Arbor, Michigan: Association for Computational Linguistics), 65–72

Belkadi, S., Micheletti, N., Han, L., Del-Pinto, W., and Nenadic, G. (2023). Generating medical instructions with conditional transformer. In *NeurIPS 2023 Workshop on Synthetic Data Generation with Generative AI*

Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*

Bengio, Y., Ducharme, R., and Vincent, P. (2000). A neural probabilistic language model. *Advances in neural information processing systems* 13

Berg, H., Henriksson, A., and Dalianis, H. (2020). The impact of de-identification on downstream named entity recognition in clinical text. In *Proceedings of the 11th International Workshop on Health Text Mining and Information Analysis*. 1–11

Berg, H., Henriksson, A., Fors, U., and Dalianis, H. (2021). De-identification of clinical text for secondary use: Research issues. In *14th International Joint Conference on Biomedical Engineering Systems and Technologies-BIOSTEC 2021, 11-13 Februar, 2021, Vienna, Austria* (SciTePress), 592–599

Boonstra, M. J., Weissenbacher, D., Moore, J. H., Gonzalez-Hernandez, G., and Asselbergs, F. W. (2024). Artificial intelligence: revolutionizing cardiology with large language models. *European Heart Journal* 45, 332–345

Bose, P., Srinivasan, S., Sleeman IV, W. C., Palta, J., Kapoor, R., and Ghosh, P. (2021). A survey on recent named entity recognition and relationship extraction techniques on clinical texts. *Applied Sciences* 11, 8319

Brown, P. F., Cocke, J., Della Pietra, S. A., Della Pietra, V. J., Jelinek, F., Lafferty, J., et al. (1990). A statistical approach to machine translation. *Computational linguistics* 16, 79–85

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., et al. (2020). Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA: Curran Associates Inc.), NIPS '20

Bundschus, M., Dejori, M., Stetter, M., Tresp, V., and Kriegel, H.-P. (2008). Extraction of semantic biomedical relations from text using conditional random fields. *BMC bioinformatics* 9, 1–14

Cai, P.-X., Fan, Y.-C., and Leu, F.-Y. (2022). Compare encoder-decoder, encoder-only, and decoder-only architectures for text generation on low-resource datasets. In *Advances on Broad-Band Wireless Computing, Communication and Applications: Proceedings of the 16th International Conference on Broad-Band Wireless Computing, Communication and Applications (BWCCA-2021)* (Springer), 216–225

Camacho-Collados, J. and Pilehvar, M. T. (2018). From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research* 63, 743–788

Cawsey, A. J., Webber, B. L., and Jones, R. B. (1997). Natural language generation in health care. *Journal of the American Medical Informatics Association* 4, 473–482

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014). Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Association for Computational Linguistics), 1724

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of machine learning research* 12, 2493–2537

[Dataset] Context.ai (2024). Compare llama 3 70b instruct to gpt-3.5 turbo. [Online; accessed 14-August-2024]

Dernoncourt, F., Lee, J. Y., Uzuner, O., and Szolovits, P. (2017). De-identification of patient notes with recurrent neural networks. *Journal of the American Medical Informatics Association* 24, 596–606

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, eds. J. Burstein, C. Doran, and T. Solorio (Minneapolis, Minnesota: Association for Computational Linguistics), 4171–4186. doi:10.18653/v1/N19-1423

DuBay, W. H. (2004). The principles of readability. *Online submission*

Eddy, S. R. (1996). Hidden markov models. *Current opinion in structural biology* 6, 361–365

Ellershaw, S., Tomlinson, C., Burton, O. E., Frost, T., Hanrahan, J. G., Khan, D. Z., et al. (2024). Automated generation of hospital discharge summaries using clinical guidelines and large language models. In *AAAI 2024 Spring Symposium on Clinical Foundation Models*

[Dataset] Eric, L. and Johnson, A. (2023). Clinical-T5: Large Language Models Built Using MIMIC Clinical Text. PhysioNet. doi:10.13026/rj8x-v335

Gatt, A. and Krahmer, E. (2018). Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research* 61, 65–170

Gillioz, A., Casas, J., Mugellini, E., and Abou Khaled, O. (2020). Overview of the transformer-based models for nlp tasks. In *2020 15th Conference on computer science and information systems (FedCSIS)* (IEEE), 179–183

Goldberger, A. L., Amaral, L. A. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., et al. (2000). Physiobank, physiotoolkit, and physionet. *Circulation* 101, e215–e220. doi:10.1161/01.CIR.101.23.e215

Graves, A. and Graves, A. (2012). Long short-term memory. *Supervised sequence labelling with recurrent neural networks* , 37–45

Grishman, R. and Sundheim, B. M. (1996). Message understanding conference-6: A brief history. In *COLING 1996 volume 1: The 16th international conference on computational linguistics*

Hirst, G., DiMarco, C., Hovy, E., and Parsons, K. (1997). Authoring and generating health-education documents that are tailored to the needs of the individual patient. In *User Modeling: Proceedings of the Sixth International Conference UM97 Chia Laguna, Sardinia, Italy June 2–5 1997* (Springer), 107–118

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation* 9, 1735–1780

Hou, L., Pang, R. Y., Zhou, T., Wu, Y., Song, X., Song, X., et al. (2022). Token dropping for efficient bert pretraining. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 3774–3784

Humbert-Droz, M., Mukherjee, P., and Gevaert, O. (2022). Strategies to address the lack of labeled data for supervised machine learning training with electronic health records: Case study for the extraction of symptoms from clinical notes. *JMIR Medical Informatics* 10, e32903

HÜSKE-KRAUS, D. (2003). Text generation in clinical medicine: A review. *Methods of information in medicine* 42, 51–60

[Dataset] International, S. (2024). Snomed ct browser. `https://browser.ihtsdotools.org/`? Accessed: 2024-09-03

[Dataset] Johnson, A., Bulgarelli, L., Pollard, T., Gow, B., Moody, B., Horng, S., et al. (2024a). Mimic-iv. doi:10.13026/hxp0-hg59

Johnson, A., Pollard, T., Horng, S., Celi, L. A., and Mark, R. (2023a). MIMIC-IV-Note: Deidentified free-text clinical notes. *PhysioNet* doi:10.13026/1n74-ne17

Johnson, A. E., Bulgarelli, L., Shen, L., Gayles, A., Shammout, A., Horng, S., et al. (2023b). Mimic-iv, a freely accessible electronic health record dataset. *Scientific data* 10, 1

Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L.-w. H., Feng, M., Ghassemi, M., et al. (2016). Mimic-iii, a freely accessible critical care database. *Scientific data* 3, 1–9

Johnson, S. J., Murty, M. R., and Navakanth, I. (2024b). A detailed review on word embedding techniques with emphasis on word2vec. *Multimedia Tools and Applications* 83, 37979–38007

Kong, M., Huang, Z., Kuang, K., Zhu, Q., and Wu, F. (2022). Transq: Transformer-based semantic query for medical report generation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (Springer), 610–620

Kovačević, A., Bašaragin, B., Milošević, N., and Nenadić, G. (2024). De-identification of clinical free text using natural language processing: A systematic review of current approaches. *Artificial Intelligence in Medicine* , 102845

Kundeti, S. R., Vijayananda, J., Mujjiga, S., and Kalyan, M. (2016). Clinical named entity recognition: Challenges and opportunities. In *2016 IEEE International Conference on Big Data (Big Data)* (IEEE), 1937–1945

Lamprou, Z., Pollick, F., and Moshfeghi, Y. (2022). Role of punctuation in semantic mapping between brain and transformer models. In *International Conference on Machine Learning, Optimization, and Data Science* (Springer), 458–472

Lee, S. H. (2018). Natural language generation for electronic health records. *NPJ digital medicine* 1, 63

Li, J., Tang, T., Zhao, W. X., Nie, J.-Y., and Wen, J.-R. (2024). Pre-trained language models for text generation: A survey. *ACM Computing Surveys* 56, 1–39

Li, Y., Wehbe, R. M., Ahmad, F. S., Wang, H., and Luo, Y. (2023). A comparative study of pretrained language models for long clinical text. *Journal of the American Medical Informatics Association* 30, 340–347

Liu, J., Shen, D., Zhang, Y., Dolan, W. B., Carin, L., and Chen, W. (2022). What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*. 100–114

Locke, S., Bashall, A., Al-Adely, S., Moore, J., Wilson, A., and Kitchen, G. B. (2021). Natural language processing in medicine: a review. *Trends in Anaesthesia and Critical Care* 38, 4–9

[Dataset] Loria, S. (2024). Textblob: Simplified text processing. `https://textblob.readthedocs.io/en/dev/`. Accessed: 2024-09-03

Lu, Q., Dou, D., and Nguyen, T. (2022). ClinicalT5: A generative language model for clinical text. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, eds. Y. Goldberg, Z. Kozareva, and Y. Zhang (Abu Dhabi, United Arab Emirates: Association for Computational Linguistics), 5436–5443. doi:10.18653/v1/2022.findings-emnlp.398

Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H., et al. (2022). Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in bioinformatics* 23, bbac409

Ma, L. and Zhang, Y. (2015). Using word2vec to process big text data. In *2015 IEEE International Conference on Big Data (Big Data)* (IEEE), 2895–2897

Masuko, T., Kobayashi, T., Tamura, M., Masubuchi, J., and Tokuda, K. (1998). Text-to-visual speech synthesis based on parameter generation from hmm. In *Proceedings of the 1998 IEEE International*

Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181) (IEEE), vol. 6, 3745–3748

McKeown, K., Jordan, D. A., Pan, S., Shaw, J., and Allen, B. A. (1997). Language generation for multimedia healthcare briefings. In *Fifth Conference on Applied Natural Language Processing*. 277–282

Meystre, S. M. (2015). De-identification of unstructured clinical data for patient privacy protection. In *Medical Data Privacy Handbook* (Springer). 697–716

Meystre, S. M., Ferrández, O., Friedlin, F. J., South, B. R., Shen, S., and Samore, M. H. (2014). Text de-identification for privacy protection: a study of its impact on clinical text information content. *Journal of biomedical informatics* 50, 142–150

Micheletti, N., Belkadi, S., Han, L., and Nenadic, G. (2024). Exploration of masked and causal language modelling for text generation. *CoRR* abs/2405.12630

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* 26

Nadkarni, P. M., Ohno-Machado, L., and Chapman, W. W. (2011). Natural language processing: an introduction. *Journal of the American Medical Informatics Association* 18, 544–551

Norgeot, B., Muenzen, K., Peterson, T. A., Fan, X., Glicksberg, B. S., Schenk, G., et al. (2020). Protected health information filter (philter): accurately and securely de-identifying free-text clinical notes. *NPJ digital medicine* 3, 57

[Dataset] OpenAI (2023). Gpt-3.5-turbo. `https://platform.openai.com/docs/models/gpt-3.5-turbo`. Accessed: 2025-03-26

Phan, L., Anibal, J. T., Tran, H. T., Chanana, S., Bahadroglu, E., Peltekian, A., et al. (2021). Scifive: a text-to-text transformer model for biomedical literature. *ArXiv* abs/2106.03598

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., and Manning, C. D. (2020). Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 101–108

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., et al. (2020a). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research* 21, 1–67

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., et al. (2020b). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* 21, 1–67

Rayner, H., Hickey, M., Logan, I., Mathers, N., Rees, P., and Shah, R. (2020). Writing outpatient letters to patients. *BMJ* 368

Ren, L., Belkadi, S., Han, L., Del-Pinto, W., and Nenadic, G. (2025). Beyond reconstruction: Generating privacy-preserving clinical letters. In *NAACL 2025 Workshop PrivateNLP*

Santhanam, S. (2020). Context based text-generation using lstm networks. *arXiv e-prints* , arXiv–2005

Satapathy, R., Cambria, E., and Hussain, A. (2017). *Sentiment analysis in the bio-medical domain* (Springer)

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., et al. (2024). Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*

Sharma, S., Diwakar, M., Singh, P., Singh, V., Kadry, S., and Kim, J. (2023). Machine translation systems based on classical-statistical-deep-learning approaches. *Electronics* 12, 1716

Spasic, I. and Nenadic, G. (2020). Clinical text data in machine learning: Systematic review. *JMIR medical informatics* 8, e17984

Sun, S. and Iyyer, M. (2021). Revisiting simple neural probabilistic language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 5181–5188

Sutton, C., McCallum, A., et al. (2012). An introduction to conditional random fields. *Foundations and Trends® in Machine Learning* 4, 267–373

Tang, R., Han, X., Jiang, X., and Hu, X. (2023). Does synthetic data generation of llms help clinical text mining? *ArXiv* abs/2303.04360

Tarur, S. U. and Prasanna, S. (2021). clinical case letter. *Indian Pediatr* 58, 189

[Dataset] the President and of Harvard College, F. (2023). Dbmi data portal. `https://portal.dbmi.hms.harvard.edu/`. Accessed: 2024-09-04

Tsirmpas, D., Gkionis, I., Papadopoulos, G. T., and Mademlis, I. (2024). Neural natural language processing for long texts: A survey on classification and summarization. *Engineering Applications of Artificial Intelligence* 133, 108231

Tucker, K., Branson, J., Dilleen, M., Hollis, S., Loughlin, P., Nixon, M. J., et al. (2016). Protecting patient privacy when sharing patient-level data from clinical trials. *BMC medical research methodology* 16, 5–14

van der Lee, C., Krahmer, E., and Wubben, S. (2018). Automated learning of templates for data-to-text generation: comparing rule-based, statistical and neural methods. In *Proceedings of the 11th International Conference on Natural Language Generation*. 35–45

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Red Hook, NY, USA: Curran Associates Inc.), NIPS'17, 6000–6010

Wang, G., Liu, X., Ying, Z., Yang, G., Chen, Z., Liu, Z., et al. (2023). Optimized glycemic control of type 2 diabetes with reinforcement learning: a proof-of-concept trial. *Nature Medicine* 29, 2633–2642

Wu, Y. (2024). Large language model and text generation. In *Natural Language Processing in Biomedicine: A Practical Guide* (Springer). 265–297

Xie, Q., Chen, Q., Chen, A., Peng, C., Hu, Y., Lin, F., et al. (2024). Me-llama: Foundation large language models for medical applications. *Research Square* , rs–3

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems* 32

Zeng, W., Jin, S., Liu, W., Qian, C., Luo, P., Ouyang, W., et al. (2022). Not all tokens are equal: Human-centric visual analysis via token clustering transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11101–11111

Zhang, Y., Zhang, Y., Qi, P., Manning, C. D., and Langlotz, C. P. (2021). Biomedical and clinical English model packages for the Stanza Python NLP library. *Journal of the American Medical Informatics Association*

Zhang, Z., Wu, Y., Zhao, H., Li, Z., Zhang, S., Zhou, X., et al. (2020). Semantics-aware bert for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 34, 9628–9635

Zhuang, L., Wayne, L., Ya, S., and Jun, Z. (2021). A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, eds. S. Li, M. Sun, Y. Liu, H. Wu, K. Liu, W. Che, S. He, and G. Rao (Huhhot, China: Chinese Information Processing Society of China), 1218–1227