# Learning Faster with Better Tokens: Parameter-Efficient Vocabulary Adaptation for Specialized Text Summarization

**Gunjan Balde[1], Soumyadeep Roy[2], Mainack Mondal[1] and Niloy Ganguly[1]**
[1]Dept. of Computer Science and Engg., IIT Kharagpur, Kharagpur, India
[2]Dept. of Medicine (Biomedical Informatics), Stanford University, Stanford, CA, USA
**Correspondence:** balde.gunjan0812@gmail.com

## Abstract

Large language models pretrained on general-domain corpora often exhibit tokenization inefficiencies when applied to specialized domains. Although continual pretraining for domain adaptation partially alleviate performance degradation, it does not resolve the fundamental vocabulary mismatch. To address this gap, we introduce a targeted parameter-efficient domain adaptation approach that combines vocabulary adaptation with pretraining for LLM-based text summarization. Our unified framework augments pretrained tokenizers with domain-specific tokens while selectively replacing under-trained and unreachable tokens to limit parameter growth. We evaluate our approach on Llama-3.1-8B and Qwen2.5-7B across legal and medical summarization tasks on a challenge-oriented evaluation protocol focused on expert-driven text and summaries which typically has higher concentration of *over-fragmented* Out-of-Vocabulary (**OOV**) words. The vocabulary adaptation algorithm enhances the overall quality of the summarization model by improving semantic similarity between the generated summaries and their references. In addition, the adapted model produces summaries that incorporate more appropriate novel and domain-specific words, leading to improved coherence, relevance, and faithfulness. We further observe that our proposed approach significantly reduce training time by $35-55\%$ over continual pretraining and reduce parameter counts up to $37\%$ w.r.t expansion-only methods. We make codebase publicly available [1].

## 1 Introduction

While large language models (LLMs) have revolutionized natural language processing, adapting generalist models to expert domains remains challenging due to high vocabulary mismatch between general and domain-specific corpora. Recent domain-specific models including Meditron-70B (Chen et al., 2023); BioMistral (Labrak et al., 2024), built on Mistral-7B and further pretrained on PubMed Central; and PMC-LLaMA (Wu et al., 2024) demonstrate that continued pretraining on specialized corpora yields substantial performance improvements. However, vocabulary mismatch fundamentally limits these gains: PubMedBERT (Gu et al., 2021) demonstrates that medical terms like "naloxone" fragment into meaningless subwords ("nal", "##ox", "##one"), while domain-specific vocabularies treat them atomically. This tokenization inefficiency imposes substantial costs—non-English and domain-specific text can require up to $13\times$ more tokens than English (Rust et al., 2021; Ahia et al., 2023; Petrov et al., 2023), directly increasing API costs, latency, and memory requirements. Recent work establishes that this fragmentation reduces effective context window size and impedes learning meaningful representations (Hofmann et al., 2022; Kaplan et al., 2025).

The conventional approach to addressing vocabulary mismatch involves domain-adaptive pretraining (DAPT), where models undergo continued pretraining on domain-specific corpora (Gururangan et al., 2020). While effective, this paradigm presents significant practical limitations. BioMistral-7B required 32 A100 GPUs for 20 hours on 3 billion tokens from PubMed Central, while Meditron-70B consumed 128 A100 GPUs for 332 hours processing 46 billion tokens while achieving marginal improvements. For contemporary large language models, Hu et al. (2022) note that full fine-tuning is "prohibitively expensive", requiring complete parameter updates and storage of separate model instances per domain. While parameter-efficient methods reduce trainable parameters, they do not address the underlying tokenization inefficiency, as the vocabulary remains unchanged.

[1] https://github.com/gb-kgp/VocabReplace-Then-Expand



*Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 45071–45086

An alternative paradigm that directly addresses vocabulary mismatch is vocabulary adaptation, modifying a pretrained model's tokenizer and embedding layer to incorporate domain-specific vocabulary. Recent works (Sachidananda et al., 2021; Hong et al., 2021; Liu et al., 2023; Yamaguchi et al., 2024; Balde et al., 2024; Gao et al., 2024; Balde et al., 2025) establishes this as a resource-efficient path. However, vocabulary expansion introduces computational overhead through parameter growth: adding 10,000 tokens to Llama-3-8B requires approximately 80 million additional parameters (at 4096-dimensional embeddings), representing non-trivial increase in model size and inference cost. Land and Bartolo (2024) reveal a critical insight: contemporary language models contain $0.1 - 1\%$ severely under-trained "glitch tokens"—vocabulary tokens that occupy vocabulary slots but contribute minimally due to insufficient pretraining exposure. This observation suggests efficient vocabulary adaptation is possible by strategically replacing under-trained tokens with domain-specific vocabulary, achieving adaptation benefits with minimal parameter expansion (Purason et al., 2026).

In this work, we propose a vocabulary adaptation method that strategically replaces under-trained and unreachable tokens with domain-specific vocabulary before resorting to expansion, thereby minimizing parameter overhead while enabling effective domain specialization. Our approach operates on Llama-3.1-8B and Qwen2.5-7B and consists of four key steps: (1) we train a BPE tokenizer on domain-specific corpora to identify candidate domain vocabulary, (2) we select the top 10,000 tokens based on frequency and coverage statistics as our vocabulary adaptation budget, (3) we compile a replacement candidate list by identifying under-trained and unreachable tokens using Land and Bartolo (2024) methodology, and (4) we replace tokens from this candidate list with domain-specific tokens, expanding the vocabulary only when the replacement budget is exhausted. This hybrid replacement-then-expansion strategy enables us to prioritize recycling underutilized vocabulary slots, minimizing net parameter increase while maximizing domain vocabulary coverage.

Beyond standard benchmarking, we introduce a challenge-oriented evaluation framework that stress-tests model performance under conditions where domain vocabulary knowledge is critical. We restructure the downstream domain-specific corpus to explicitly capture challenging scenarios: test sets with high out-of-vocabulary (OOV) concentrations in either source documents (SD) and reference summaries (RS)–***OOV_SD*** and ***OOV_RS*** respectively. We also take a ***Random*** subset without any restriction on OOV concentration to compare the degree of performance in these challenging scenarios. This targeted evaluation approach allows us to assess how well generalist models handle expert-level summarization tasks where domain-specific terminology is essential, providing a more rigorous test of vocabulary adaptation effectiveness beyond aggregate performance metrics. We evaluate our approach on two specialized domains—*medical and legal literature*—demonstrating that our method achieves competitive or superior performance compared to conventional vocabulary extension while substantially reducing parameter overhead and maintaining inference efficiency.

We hypothesize that the effectiveness of vocabulary adaptation is governed by the severity and location of lexical mismatch between pretrained tokenizers and downstream data. We find that: (i) across challenging scenarios of ***OOV_SD*** and ***OOV_RS***, we observe more improvement in former setting over competing baselines. Although margins of gain in slightly higher in ***OOV_RS*** ($4.44\%$) than ***OOV_SD*** ($4.26\%$); (ii) performance gains are notably higher than the gains observed in ***Random*** setting ($3.06\%$), validating that gains are higher in higher vocabulary mismatch scenario; (iii) vocabulary adaptation enables models to reach their best-performing checkpoints **35–55% earlier** than continual pretraining alone, reducing the training time; (iv) hybrid replacement-then-expansion strategy remains highly parameter-efficient reducing parameters by $12.04\%$ and $37.19\%$ for Llama and Qwen models respectively averaged across both the domains. These results identify tokenization mismatch as a bottleneck in domain adaptation and motivate vocabulary-adaptation strategies as a targeted, data-dependent intervention. We make our codebase publicly available at `https://github.com/gb-kgp/VocabReplace-Then-Expand`.

## 2 Proposed Methodology (VOCABADAPT)

### 2.1 Background

Generalist LLMs are pretrained on broad-coverage corpora, resulting in tokenizers optimized for general text distributions. When deployed on specialized domains such as medical text, these tokeniz-

ers exhibit systematic over-fragmentation. For instance, the term "Osteoporosis" is tokenized as `[O, ste, opor, osis]` by the Llama tokenizer, splitting into four subwords. This over-fragmentation introduces two primary challenges: first, the model must reconstruct semantic meaning across multiple token positions, increasing computational overhead and representation noise; second, generation becomes error-prone as the model must correctly predict each fragment in sequence, with errors compounding across token boundaries.

The standard solution to this vocabulary mismatch problem involves expanding the model's vocabulary by adding domain-specific tokens. Let $V_{\text{src}}$ denote the source vocabulary of size $|V_{\text{src}}|$ with corresponding embedding matrix $E \in \mathbb{R}^{|V_{\text{src}}| \times d}$ and unembedding matrix $U \in \mathbb{R}^{d \times |V_{\text{src}}|}$, where $d$ represents the model's hidden dimension. Adding $k$ domain-specific tokens to form an expanded vocabulary $V_{\text{exp}} = V_{\text{src}} \cup V_{\text{new}}$ necessitates expanding both embedding and unembedding matrices, introducing $2k \cdot d$ additional parameters. For models with large hidden dimensions and substantial domain vocabularies, this parameter overhead becomes significant, increasing memory footprint and inference cost.

We propose an alternative approach that challenges the necessity of vocabulary expansion. Our central hypothesis is that generalist tokenizers contain a substantial subset of undertrained and unreachable tokens that contribute minimally to model performance. Rather than expanding the vocabulary, we identify these ineffectual tokens and replace them with domain-specific terminology, maintaining constant vocabulary size while addressing fragmentation. When domain requirements exceed the available candidate tokens, we resort to expansion only for the remaining terms, thereby minimizing parameter growth.

### 2.2 Identifying Candidate Tokens for Replacement

Our replacement strategy relies on identifying a candidate set $V_{\text{cand}} \subseteq V_{\text{src}}$ comprising tokens that satisfy two independent criteria: they must be undertrained and unreachable.

The undertrained tokens are identified through the methodology of Land and Bartolo (2024) where the L2 norm for each token embedding $e_i$ in the vocabulary is computed, $\|e_i\|_2$, excluding partial utf-8, fallback bytes, and unreachable tokens. Their analysis demonstrates that tokens with embedding norms below a threshold corresponds to vocabulary items that appeared infrequently during pretraining and hence undertrained. This token token set is henceforth represented as $V_{\text{undertrained}}$.

The unreachable tokens are identified through a consistency test (Land and Bartolo, 2024; Purason et al., 2026). A token $t$ is deemed unreachable if decoding[2] its corresponding vocabulary token-id $t_i$ and encoding the decoded token does not yield the original token-id $t_i$. E.g. decoding the encoding token-id 378 in Llama-3.1-8B results in âĢ, which upon encoding yield token-id 5809. Formally, a token is unreachable when $\text{encode}(\text{decode}(t_i)) \neq [t_i]$. These tokens represent vocabulary entries that cannot be produced through the standard tokenization algorithm and thus remain inaccessible during normal model inference. While they occupy vocabulary slots and contribute to parameter count, they serve no functional role in model operation. This token set is henceforth represented as $V_{\text{unreachable}}$.

We define our candidate set as the union of these two criteria:

$$V_{\text{cand}} = V_{\text{undertrained}} \cup V_{\text{unreachable}} \tag{1}$$

This union ensures we replace tokens that are poorly trained and inaccessible, providing a conservative strategy that minimizes risk of degrading model performance on general domains. Empirically, we observe that approximately $3 - 4\%$ percent of vocabulary tokens in both Llama-3.1-8B and Qwen2.5-7B satisfy the candidate set criterion, providing a substantial pool of replacement candidates.

We apply a final refinement to ensure tokenizer integrity. BPE (Byte-Pair Encoding) subword tokenization algorithm construct vocabulary through iterative merge operations, where character sequences are progressively combined into larger units based on merge rules. Replacing a token that appears in the merge rule of another token outside the candidate set would fundamentally break the tokenization process, rendering certain vocabulary tokens untokenizable. To prevent this, we filter the candidate set to exclude any token that appears as a component in the merge rule of a token not designated for replacement. We construct a directed acyclic graph (DAG) with nodes as the token-id

[2] encoding and decoding here corresponds to buit-in `tokenizer.encode` and `tokenizer.decode` function calls of a model tokenizer.

and an edge from token-i to token-j marking the relationship if token-i contributed in merge-rule of token-j (E.g., in → ing). Then, for every candidate that could be replaced, we checked if it has any descendants (nodes reachable from this node) that lies outside the candidate replacement set. If yes, we do not replace it, else we consider it for replacement. This set of tokens is marked as $V_{\text{exclude}}$. This constraint guarantees that all remaining merge rules remain valid after vocabulary modification, preserving the deterministic and complete nature of the tokenization algorithm. The refined candidate set therefore contains only tokens that are undertrained, unreachable, and removing does not compromise the structural integrity of the tokenizer.

$$V_{\text{cand}} = V_{\text{cand}} \backslash V_{\text{exclude}} \tag{2}$$

The final replacement candidate set is of size 1528 for Llama-3.1-8B (vocabulary size: 128K) and 3987 for Qwen-2.5-7B (vocabulary size: 151K). We next describe our domain-specific vocabulary construction step.

### 2.3 Building Domain-Specific Vocabulary

We construct domain-specific vocabulary through a process involving corpus curation, independent tokenizer training, and vocabulary filtering for each target domain. This approach ensures that our added tokens genuinely represent domain-salient terminology rather than arbitrary subword fragments.

We curate two domain-specific corpora, each comprising 100 million tokens (100M) sampled from authoritative sources within their respective domains. The medical domain corpus is sampled from the MEDITRON pretraining corpora (Chen et al., 2023), which aggregates clinical practice guidelines, PubMed Central full-text articles, and article abstracts, providing comprehensive coverage of both clinical and biomedical language. For the legal domain, we compile a corpus from Supreme Court of India case documents, capturing the specialized vocabulary and linguistic conventions of Indian jurisprudence.

We train an independent Byte-Pair Encoding tokenizer using the HuggingFace tokenizers [3] library with a vocabulary size of 256,000 tokens dor each domain corpus. This training process learns domain-optimized merge operations that naturally surface frequently occurring domain-specific terms as single tokens. From each trained domain tokenizer vocabulary, we extract candidate tokens for addition to the base model. We filter this set to exclude any tokens that already exist in the source model vocabulary $V_{\text{src}}$, as these tokens require no adaptation. This non-overlapping constraint ensures we only add genuinely new vocabulary items that address coverage gaps in the original tokenizer.

We apply an additional refinement to ensure linguistic coherence across models and avoid introducing problematic tokens. We restrict the candidate set to tokens containing only English alphabetic characters, excluding any subwords that contain numeric digits, special symbols, or mixed alphanumeric patterns. This filtering serves multiple purposes: it eliminates formatting artifacts, date fragments, and identifier components that do not represent meaningful linguistic units; it ensures that added tokens correspond to genuine lexical items rather than incidental character sequences; and it maintains consistency with the predominantly alphabetic nature of established vocabulary in pretrained models. The resulting filtered set forms our domain-specific vocabulary $V_{\text{new}}^{\mathcal{D}}$, comprising high-frequency, domain-salient, purely alphabetic tokens that address the most significant tokenization inefficiencies for the target domain.

In both the settings, we select the top 10,000 vocabulary tokens ranked by frequency in the domain corpus, representing the most salient domain-specific vocabulary items. We next describe the procedure of vocabulary replacement.

### 2.4 Vocabulary Replacement-Then-Expansion and Embedding Initialization

Thus far, we have a domain vocabulary $V_{\text{new}}^{\mathcal{D}}$ and replacement candidate set $V_{\text{cand}}$(Eq. 2), such that $|V_{\text{new}}^{\mathcal{D}}| > |V_{\text{cand}}|$ . We first replace the $V_{\text{cand}}$ from LLM's base vocabulary with equal sized set from $V_{\text{new}}^{\mathcal{D}}$ sorted by the natural merge order. We then expand the base vocabulary with the remaining $|V_{\text{new}}^{\mathcal{D}}| - |V_{\text{cand}}|$ elements from $V_{\text{new}}^{\mathcal{D}}$.

Initializing embeddings for the newly replaced and added tokens presents a critical challenge, as random initialization would require substantial training to achieve reasonable representations. Instead, we employ subword aggregation (Yamaguchi et al., 2024), leveraging model's existing understanding of subwords. For each new token $t_{\text{new}}$, we tokenize it using the original tokenizer to obtain a sequence of source tokens $[t_1, \ldots, t_n]$. We then initialize the new token's embedding as the

[3] https://github.com/huggingface/tokenizers

mean of these constituent embeddings:

$$e_{t_{\text{new}}} = \frac{1}{n} \sum_{i=1}^{n} e_{t_i} \tag{3}$$

This initialization provides a reasonable starting point that captures compositional semantics while allowing subsequent training to refine the representation. The same subword aggregation strategy is applied to initialize the corresponding unembedding matrix row. Next, we describe the procedure to tune the model with the modified vocabulary.

### 2.5 Domain-Specific Continual Pretraining

Following vocabulary modification, we conduct domain-specific continual pretraining to adapt the model to the target domain while training the new token representations. We employ Low-Rank Adaptation (LoRA) (Hu et al., 2022) to enable parameter-efficient training, inserting trainable low-rank matrices into the model's attention and feed-forward layers while keeping the original pretrained parameters frozen. This approach substantially reduces the number of trainable parameters and memory requirements during adaptation.

Each domain model is trained independently on a domain-specific corpus of 100M tokens sampled from high-quality sources representative as discussed previously. We train using the standard causal language modeling objective with next-token prediction, optimizing the model to predict each token given all preceding context. Training is conducted separately for medical and legal domains, producing two specialized model variants from each base model architecture.

## 3 Experimental Setup

Here, we describe the evaluation metrics and datasets used, followed by the baseline models and implementation details.

**Datasets.** We test our pipeline on two summarization datasets one from each domain. We use the English subset of MultiClinSumm dataset (Lima López et al., 2025) for medical domain. The dataset comprises clinical case reports as source document (SD) and their corresponding summaries derived from case report as the reference summaries (RS). We use the abstractive summarization dataset (IN-ABS) proposed in Shukla et al. (2022) for Legal domain. Here SD is a court case judgment from an Indian court and RS is an abstractive summary of the case judgment. To understand the generalizability of our approach across tasks, we further supplement the evaluation for medical domain on two summarization tasks: Evidence-based summarization (Mollá and Santiago-Martinez, 2011) and patient healthcare query summarization (Ben Abacha and Demner-Fushman, 2019; Van Veen et al., 2024). The EBM (Evidence-based Summarization) comprises a query accompanied by a PubMed abstract as a context as the source document and the reference summary as answer to the question in context of the query. CHQ (Patient healthcare query summarization) consists of the a patient-written healthcare query as input and a medical-expert written one-line concise question for the patient query as the summary. In the main text we discuss the results using clinical report summarization and the results for EBM and CHQ datasets in Appendix A.

**Restructuring Datasets for Expert-Level Summaries.** We restructure the standard dataset in such a way that challenging data points constitute our test set (Balde et al., 2024, 2025). We specifically consider two scenarios where: a) the source documents have higher OOV concentration– ***OOV_SD***, b) the reference summaries have higher OOV (Out-of-Vocabulary) concentration– ***OOV_RS***. The top-10% of data points from each of these categories are considered higher concentration documents which constitute our restructured test set. The rest 90% of corpus is kept as training set. Additionally, we create an equal-sized ***Random*** train/test subset without any restrictions on OOV concentrations to understand the degree of improvements in challenging scenarios. The dataset statistics are reported in Table 1. We note that there is roughly $30 - 40\%$ overlap in the test set of challenging scenarios.

**Baseline Models.** We used the base variants of two LLMs - Qwen-2.5 (Qwen: et al., 2025) (Model id: Qwen/Qwen2.5-7B), and Llama-3.1 (Touvron et al., 2023) (Model id: meta-llama/Llama-3.1-8B) as our *BASE* models. They do not undergo vocabulary adaptation and continual pretraining. Additionally, we also used continually pretrained variants of these base models on domain-specific text, which we label as 'CPTOnly (No Vocab Adapt)'. This helps us to evaluate the improvements observed solely because of vocabulary adaptation.

| | Corpus Size | SD Token Count | | RS Token Count | | SD OOV Conc. | | RS OOV Conc. | |
|---|---|---|---|---|---|---|---|---|---|
| | | Llama | Qwen | Llama | Qwen | Llama | Qwen | Llama | Qwen |
| **Medical** | | | | | | | | | |
| **Random** | 399 | 823 | 847 | 150 | 153 | 11.83 | 11.88 | 13.51 | 13.59 |
| ***OOV_SD*** | 399 | 808 | 828 | 147 | 150 | 16.90 | 16.94 | 17.23 | 17.24 |
| ***OOV_RS*** | 399 | 837 | 862 | 137 | 139 | 14.40 | 14.43 | 21.61 | 21.65 |
| **Legal** | | | | | | | | | |
| **Random** | 711 | 5870 | 6059 | 1171 | 1221 | 4.75 | 4.76 | 4.76 | 4.76 |
| ***OOV_SD*** | 710 | 4661 | 4801 | 940 | 975 | 7.48 | 7.49 | 6.97 | 6.98 |
| ***OOV_RS*** | 711 | 5058 | 5214 | 856 | 889 | 6.39 | 6.40 | 8.57 | 8.57 |

Table 1: Dataset statistics across Legal and Medical domains under ***Random***, ***OOV_RS***, and ***OOV_SD*** settings, reporting mean token counts, OOV concentration (fraction of unigrams in text split more than once), and novel unigram concentration (fraction of unigrams in RS not present in SD). Medical domain exhibits higher OOV concentrations than Legal domain Legal domain has substantially higher token counts than Medical domain.

| **Prompt structure** |
|---|
| **Medical** |
| You are an expert medical professional.<br>####<br>Summarize the given clinical case report into a discharge summary of 100 words or less. Use the examples to guide word choice.<br>Clinical Case Report 1: {Train-Case-Document}<br>Discharge Summary 1 : {Train-Summary}<br>##<br>Clinical Case Report 2: {Test-Case-Document}<br>Discharge Summary 2 : |
| **Legal** |
| You are an expert Indian Legal professional.<br>####<br>Summarize the given legal case document in 300 words on less. Use the examples to guide word choice.<br>Case Document 1: {Train-SD}<br>Summary 1 : {Train-RS}<br>##<br>Case Document 2: {Test-SD}<br>Summary 2 : |

Table 2: The prompt structure used for prompting LLMs inspired based on the structure proposed in ClinSumm (Van Veen et al., 2024). Since we are using BASE LLMs there is no explicit segregation of system prompt and user prompt.

**Training and Inference Strategy.** All the experiments are conducted on a single H100 80 GB GPU. We train the models using standard causal language modeling task of next token prediction and use greedy decoding to generate summaries. We use LoRA and set rank at 32, alpha at 64, learning rate at $2e-5$. For all the domains, we adapt a vocabulary of size $10K$ and train the models on $100M$ tokens dataset for a total of 3 epochs with an effective batch size of 64. Both CPTOnly and VOCABADAPT are trained on identical corpora and hyperparameter setting, with VOCABADAPT additionally performing a one-time vocabulary construction step that takes roughly 30 minutes (on a single core of Apple M3 Pro laptop). Despite this overhead, VOCABADAPT completes training in $6.5\check{}8.5$ hours total, making it notably faster than CPTOnly, which requires $10.5\check{}12.5$ hours. For inference, we use in-context learning (Brown et al., 2020) to provide inputs to model with only one example demonstration appended to the test data point (Appendix A.1 contains details on the sampling procedure for ICL demonstration). The prompt structure for ICL is provided in Table 2.

**Evaluation Metrics.** We evaluate the summarization quality using Rouge-LCS (R-LCS) as the main evaluation metric and report F-score values, as followed by prior works (Balde et al., 2024, 2025; Fabbri et al., 2021). We also report BertScore (Zhang et al., 2020) where we use BioBert (Lee et al., 2020) embeddings and InLegalBERT (Paul et al., 2023) embeddings for the medical and legal domain evaluation respectively. We also conduct a LLM-as-judge evaluation of the summaries generated in medical and legal domains. We use the Google's MedGemma-27B model (Sellergren et al., 2025) for medical domain and Gemma3-27B (Team et al., 2025) for legal domain to evaluate the model-generated summaries across three evaluation dimensions: coherence, relevance, and faithfulness on a scale of $1-5$ (Fabbri et al., 2021; Zhang et al., 2023).

## 4 Experimental Results

We report Rouge-LCS, BERTScore, and Fragment Scores (avg. number of subwords a word is tokenized into) in Table 3 focusing on best vocabulary adaptation strategies. Further results are provided in Appendix A. We observe that the impact of vocabulary expansion is strongly domain-dependent. Improvements are more pronounced in medical domain which has higher OOV concentration as compared to legal domain. We now provide a detailed discussion across scenarios highlighting where vocabulary adaptation does and does not work.

**Vocabulary adaptation leads to a lower fragment score.** Vocabulary adaptation techniques improve fragment score, thus reducing over-fragmentation and addressing vocabulary mismatch. In medical domain, we see a reduction

| | Best | *Random* | | | | *OOV_SD* | | | | *OOV_RS* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Ckpts. | $\text{FrSr}_{\text{SD}}$ ↓ | $\text{FrSr}_{\text{RS}}$ ↓ | R-LCS ↑ | BSr ↑ | $\text{FrSr}_{\text{SD}}$ ↓ | $\text{FrSr}_{\text{RS}}$ ↓ | R-LCS ↑ | BSr ↑ | $\text{FrSr}_{\text{SD}}$ ↓ | $\text{FrSr}_{\text{RS}}$ ↓ | R-LCS ↑ | BSr ↑ |
| **Medical** | | | | | | | | | | | | | |
| Llama-3.1-8B-BASE | - | 1.16 | 1.16 | 23.03 | 70.66 | 1.25 | 1.27 | 24.39 | 71.35 | 1.20 | 1.34 | 21.33 | 70.41 |
| CPTOnly (No Vocab Adapt) | 7500 | 1.16 | 1.16 | 24.89 | 75.55 | 1.25 | 1.25 | 26.29 | 76.22 | 1.20 | 1.34 | **23.92** | 75.43 |
| VOCABADAPT | 3500 | **1.05** | **1.06** | **24.98** | **75.98** | **1.09** | **1.09** | **26.68** | **76.55** | **1.06** | **1.12** | 23.65 | **75.58** |
| Qwen2.5-7B-BASE | - | 1.19 | 1.23 | 15.51 | 43.35 | 1.28 | 1.29 | 14.15 | 37.44 | 1.24 | 1.36 | 12.23 | 35.22 |
| CPTOnly (No Vocab Adapt) | 8000 | 1.19 | 1.23 | 24.73 | 75.32 | 1.28 | 1.29 | 25.96 | 75.90 | 1.24 | 1.36 | 22.41 | 74.44 |
| VOCABADAPT | 3500 | **1.09** | **1.10** | **25.04** | **75.72** | **1.12** | **1.11** | **26.11** | **76.15** | **1.10** | **1.14** | **23.12** | **75.21** |
| **Legal** | | | | | | | | | | | | | |
| Llama-3.1-8B-BASE | - | 1.03 | 1.03 | **25.89** | 67.04 | 1.08 | 1.06 | 24.82 | 64.10 | 1.06 | 1.08 | **24.86** | 65.28 |
| CPTOnly (No Vocab Adapt) | 10000 | 1.03 | 1.03 | 25.36 | 69.05 | 1.08 | 1.06 | 24.83 | 68.04 | 1.06 | 1.08 | 24.36 | **68.38** |
| VOCABADAPT | 6500 | **1.01** | **1.01** | 25.42 | **69.14** | **1.01** | **1.01** | **24.89** | **68.12** | **1.01** | **1.01** | 23.92 | 68.11 |
| Qwen2.5-7B-BASE | - | 1.06 | 1.06 | 10.63 | 28.25 | 1.11 | 1.10 | 9.97 | 27.64 | 1.09 | 1.12 | 9.90 | 27.15 |
| CPTOnly (No Vocab Adapt) | 10500 | 1.06 | 1.06 | **25.68** | **69.04** | 1.11 | 1.10 | **25.16** | **67.60** | 1.09 | 1.12 | **24.72** | **67.97** |
| VOCABADAPT | 6500 | **1.02** | **1.02** | 24.23 | 67.89 | **1.05** | **1.05** | 23.60 | 66.19 | **1.04** | **1.05** | 23.22 | 66.89 |

Table 3: Comparison of best vocabulary adaptation methods across different domains (Legal and Medical) using in-context learning with one exemplar demonstration in two challenging scenarios –***OOV_RS*** and ***OOV_SD*** and ***Random*** subset. We report Rouge-LCS (**R-LCS**), BERTScore (**BSr**), and Fragment scores in SD ($\textbf{FrSr}_{\textbf{SD}}$) and RS ($\textbf{FrSr}_{\textbf{RS}}$). We note that: (i) vocabulary adaptation significantly brings down fragment scores, (ii) improvement margins are higher in medical domain compared to legal domain (owing to higher OOV concentration), (iii) improvements due to vocabulary adaptation is typically higher in challeneging scenarios than ***Random*** setting. (iv) vocabulary adaptation brings down training time by $35-55\%$ compared to CPTOnly baselines. This contrast between ***OOV_SD*** and ***OOV_RS*** highlights that source-side OOV primarily affects content understanding, while reference-side OOV impacts lexical realization, and effective vocabulary adaptation is crucial in addressing both challenges beyond what is observed in the ***Random*** setting.

of $16.02\%$ and $15.63\%$ for Llama and Qwen respectively across challenging OOV scenarios. In legal domain, we see a reduction of $5.95\%$ and $5.73\%$ for Llama and Qwen respectively across challenging scenarios. This reduction makes models energy-efficient as fewer tokens are needed to encode and generate compared to BASE, resulting in better representations.

**Vocabulary Adaptation improves more in OOV concentration subset of source document versus reference summary.** Vocabulary adaptation improves in all cases (in terms of R-LCS and BERTScore) over BASE and 6 out of 8 comparisons over CPTOnly in ***OOV_SD***. In ***OOV_RS***, vocabulary adaptation improves in a total of 7 out of 8 comparisons over BASE and only 3 out of 8 comparisons over CPTOnly. The observed improvement can be attributed to a greater reduction in source-side token fragmentation — $10.16\%$ for ***OOV_SD*** compared to $8.92\%$ for ***OOV_RS***. Higher fragmentation in ***OOV_RS*** leads to a more dispersed attention distribution, which can hinder the model's ability to effectively capture and understand the source document, ultimately affecting overall performance.

**Improvements in medical domain is higher than legal domain.** Although for both the domains vocabulary adaptation has consistently improved over BASE. This behavior could be tied to rather simple observation from Table 1. Medical domain has substantially higher OOV concentrations in source documents and reference summaries which make it an ideal candidate for vocabulary adaptation.

**Improvement in *Random* is moderate compared to *OOV* settings.** ***Random*** setting yields slightly lower absolute performance than challenging OOV scenarios for medical domain (Qwen: 75.72 vs ***OOV_SD*** 76.15 BSr; Llama: 75.98 vs ***OOV_SD*** 76.55 BSr), validating that vocabulary adaptation is most beneficial under severe OOV constraints. The performance gap between ***Random*** and OOV scenarios is more pronounced in medical domain, consistent with higher OOV concentration in SD and RS subsets. Fragmentation reduction is more substantial in OOV scenarios than ***Random*** for medical domain (Qwen: FrSr 1.10-1.14 in OOV vs 1.09-1.10 in Random), demonstrating VOCABADAPT higher performance is consistent with higher reduction in fragmentation. That said, it needs to be mentioned that in a random situation VOCABADAPT has positive impact albeit small.

**Vocabulary adaptation improves training efficiency.** Beyond final performance, vocabulary adaptation methods consistently achieve their best checkpoints substantially earlier than CPTOnly across domains and model families. Concretely, from Table 3, we note that vocabulary adaptation variants results in an approximate **35–55% reduction in training steps** to peak performance compared to CPTOnly.This in turn reduce training time while maintaining similar performance or even outperforming CPTOnly. This indicates that correct-

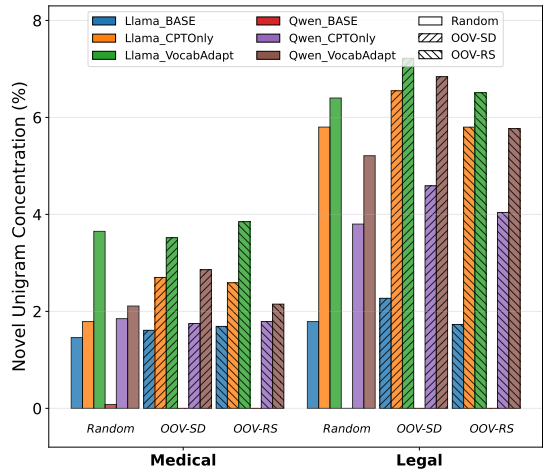



Figure 1: Median novel unigram concentration observed in the summaries generated by BASE, CPTOnly, and VOCABADAPT methods for Llama and Qwen models. We note that vocabulary adaptation method brings more (meaningful) novel words compared to baselines.

ing tokenization mismatch improves optimization efficiency by allowing models to allocate capacity to coherent domain tokens.

**Vocabulary adaptation improves semantic overlap.** We conduct a brief analysis to understand why in certain cases there is a slight drop in Rouge-LCS but gains in BERTScore. We hypothesize that since vocabulary adaptation results in higher abstraction–generation of novel unigrams that are absent in the source document is more prevalent. This might in turn brings terms that are not lexically overlapping with reference summary but does carry similar semantics. We report our findings in Figure 1. We note that vocabulary adaptation indeed introduce more meaningful abstraction (novel unigrams) than baselines consistently across evaluation scenarios. Thus validating it might account for slight drop in Rouge-LCS complemented by increase in BERTScore. The next question which can be asked is whether introduction of such novel words improve the readability, coherence of the summary, we answer that question by using LLM as a judge.

**LLM-as-a-Judge Evaluation for measuring quality of a generated summary.** We conduct a LLM-as-a-Judge evaluation (Croxford et al., 2025) of the summaries generated by CPTOnly and vocabulary adaption methods in medical and legal domain. We conduct evaluation across three dimensions of coherence, relevance, and faithfulness as done in prior art (Zhang et al., 2023; Balde et al., 2024, 2025). We take 100 random samples from medical domain and 20 from legal domain distributed uniformly across OOV scenarios and models. We report the average scores in Table 4. We find that vocabulary adaptation generates more coherent, relevant, and faithful summaries compared to competitive CPTOnly baseline. (See Appendix A.4 for further details).

| | Coherence | Relevance | Faithfulness |
|---|---|---|---|
| **Medical** | | | |
| Llama-CPTOnly (No Vocab Adapt) | 2.70 | 3.30 | 3.48 |
| Llama-VOCABADAPT | **3.30** | **3.82** | **3.84** |
| Qwen-CPTOnly (No Vocab Adapt) | 2.56 | 3.34 | 3.76 |
| Qwen-VOCABADAPT | **3.24** | **3.92** | **3.94** |
| **Legal** | | | |
| Llama-CPTOnly (No Vocab Adapt) | 2.70 | 2.80 | 2.30 |
| Llama-VOCABADAPT | **2.70** | **3.60** | **3.80** |
| Qwen-CPTOnly (No Vocab Adapt) | **2.60** | 1.90 | 3.60 |
| Qwen-VOCABADAPT | 2.20 | **2.50** | **4.30** |

Table 4: LLM-as-a-Judge results for Medical domain using MedGemma-27B, and Gemma-27B model for Legal domin as the evaluator. The evaluation is carried out across coherence, relevance, and faithfulness on a scale of $1-5$. We observe that the summaries generated by vocabulary adaptation methods are mostly rated higher than CPTOnly baseline, resulting in better summaries.

| | Vocab. Size | Params Increment | *OOV_SD* R-LCS | *OOV_SD* BSr | *OOV_RS* R-LCS | *OOV_RS* BSr |
|---|---|---|---|---|---|---|
| **Medical** | | | | | | |
| **23.92** | 75.43 | | | | | |
| Llama-VOCABADAPT w/o Replace | 141791 | 110M | 26.56 | 76.52 | 23.86 | 75.54 |
| Llama-VOCABADAPT w/ Replace | 140263 | 98M | **26.68** | **76.55** | 23.65 | **75.58** |
| Qwen-VOCABADAPT w/o Replace | 162738 | 79M | 26.11 | 76.15 | **23.12** | **75.21** |
| Qwen-VOCABADAPT w/ Replace | 158751 | 50M | **26.43** | **76.19** | 23.06 | 74.98 |
| **Legal** | | | | | | |
| Llama-VOCABADAPT w/o Replace | 139470 | 91M | 24.17 | 67.45 | 23.71 | 67.64 |
| Llama-VOCABADAPT w/ Replace | 137942 | 79M | **24.89** | **68.12** | **23.92** | **68.11** |
| Qwen-VOCABADAPT w/o Replace | 162352 | 77M | 23.08 | 66.04 | 22.26 | 66.13 |
| Qwen-VOCABADAPT w/ Replace | 158365 | 48M | **23.60** | **66.19** | **23.22** | **66.89** |

Table 5: Ablation analysis for vocabulary adaptation methods with and without replacement. We show vocabulary sizes, parameter counts (in Millions), and performance metrics, R-LCS and BERTScore in challenging scenarios. We note that replacement-based methods (i) save $12.04\%$ parameters in Llama-3.1 and $37.19\%$ parameters in Qwen2.5-7B, (ii) performs better than without replacement in $13/16$ settings.

**Ablation analysis of vocabulary adaptation with and without replacement.** We report an ablation of vocabulary adaptation techniques with and without replacement in Table 5. We note that replacement-based strategies perform better or at par with without-replacement strategies in 13 out of 16 settings. Replacement based strategies favors Llama (7 out of 8 settings) slightly more than Qwen (6 out of 8 settings). Contrary to previous discussions on higher OOV concentration in medical domain, we find here that Legal domain benefits more (all 8 settings) than medical domain (5 out of 8 settings). One possible explanation for this observation could be higher replacement fraction

in legal domain (25.43%) compared to medical domain (23.81%). Importantly, replacement-based vocabulary adaptation does not increase the number of trainable parameters beyond the expanded embedding and unembedding layer ($lm\_head$). We note that replacement-based methods save 12.04% parameters in Llama-3.1 and 37.19% parameters in Qwen2.5-7B.

**Comparison with closed-source LLMs.** We conducted a zero-shot analysis on a closed-source model: GPT-5 (`gpt-5-mini-2025-08-07`). We aim to understand that as the number of parameters increases, does over-fragmentation still persists as an underlying issue. To that end, we ran the evaluation on GPT-5 and compared the results with the best of Llama and Qwen results on VOCABADAPT method. The results are shown in Table 6.

| | Random | | *OOV_SD* | | *OOV_RS* | |
|---|---|---|---|---|---|---|
| Model | R-LCS | BSr | R-LCS | BSr | R-LCS | BSr |
| **Medical** | | | | | | |
| GPT-5-mini | 21.80 | 73.89 | 23.95 | 75.07 | 22.65 | 74.81 |
| VOCABADAPT$_{\text{Best}}$ | **24.98** | **75.98** | **26.68** | **76.55** | **23.65** | **75.58** |
| **Legal** | | | | | | |
| GPT-5-mini | 19.94 | 67.54 | 19.45 | 66.40 | 20.46 | 67.08 |
| VOCABADAPT$_{\text{Best}}$ | **25.42** | **69.14** | **24.89** | **68.12** | **23.92** | **68.11** |

Table 6: Performance of GPT-5-mini and VOCABADAPT$_{\text{Best}}$ on summarization across Medical and Legal domains, evaluated using Rouge-LCS (R-LCS) and BERTScore (BSr) under challenging scenarios (***OOV_SD*** and ***OOV_RS***) and Random setting.

We note that our 7-8B parameter model with vocabulary adaptation is consistently outperforming gpt-5-mini (speculated several orders larger than 7B model with more complex architecture and workflow including MoE imbibed) in all the scenarios. This motivates the need for vocabulary adaptation even for larger parameter models.

## 5 Related Works

**Domain Adaptation via Continued Pretraining.** Standard adaptation strategies rely on continued pretraining (CPT) to align generalist models to expert domain. Prominent examples include MEDITRON (Chen et al., 2023) BioMistral (Labrak et al., 2024) and ChatLaw (Cui et al., 2024), which utilize massive domain corpora to enhance performance. However, these model-centric approaches are computationally-intensive and fail to address the underlying tokenization over-fragmentation (Si et al., 2019), leading to inefficient inference and context window erosion (Gu et al., 2021).

**Vocabulary Expansion Strategies.** To mitigate fragmentation, recent research has pivoted towards vocabulary expansion. Hong et al. (2021) introduced AVocaDo to optimize vocabulary based on fragment scores (Rust et al., 2021), while Task-Adaptive Tokenization (Liu et al., 2023) leverages subword regularization to reduce sequence length. More targeted approaches like MEDVOC (Balde et al., 2024), Gold Panning (Liu et al., 2024), HYPEROFA (Özeren et al., 2025), AdaptiVocab (Nakash et al., 2025), and MEDVOC-LLM and ScafFix (Balde et al., 2025) focus on selecting high-value domain tokens, though these additive methods inevitably increase the model's parameter count and memory footprint.

**Vocabulary Pruning and Replacement.** Addressing parameter efficiency, emerging works investigates pruning and token recycling. Land and Bartolo (2024) identified "glitch tokens" as undertrained vocabulary artifacts ripe for removal. Building on this, methods like Vocab Diet (Reif et al., 2025), and COMPACT (Kong et al., 2025) demonstrate that pruning unused tokens or replacing them with domain-specific terms can maintain performance (Purason et al., 2026). This establishes the basis for our replacement-based framework, which achieves adaptation with relatively less parameter growth as compared to expansion techniques.

## 6 Conclusion

We presented a systematic study of vocabulary adaptation for domain-specific summarization, focusing on when and why it improves LLMs performance. Across controlled settings: ***OOV_RS*** and ***OOV_SD***, we showed that gains are governed by the severity and location of vocabulary mismatch. Vocabulary adapted models converge faster ($35-55\%$) than continual pretraining alone. Furthermore, vocabulary adaptation not only improves performance quantitatively (in terms of ROUGE-LCS and BERTScore) but also qualitatively (coherence, relevance, and faithfulness) as noted in our LLM-as-a-Judge evaluation. Replacement-based strategies remain parameter-efficient saving up to 37% parameters and further improve robustness over expansion-only counterpart. These findings position tokenization as a design consideration for future domain adaptation works. We make our codebase pulicly available at `https://github.com/gb-kgp/VocabReplace-Then-Expand`.

## 7 Limitations

Our work has the the following limitations. First, we built our *fixed-size* 100M pretraining corpora inspired from prior art (Beltagy et al., 2019; Chen et al., 2023; Paul et al., 2023); however, there can be many other ways to come up with a much more fine-grained pretraining corpora. This can be an interesting future work to explore. Second, we note that LLMs considered, Llama-3.1 and Qwen2.5, have large vocabulary sizes (128K and 151K); still, there is a significant overlap in the vocabulary of these models. However, this in no way affects the findings of this work. It could indeed be interesting to explore the efficacy of these strategies of other varying vocabulary size models, like Microsoft-Phi (Abdin et al., 2024) with a vocabulary size of 100K, and Gemma (Team et al., 2025) series with a vocabulary size of 256K. Third, we fix the size of expansion vocabulary at 10K based on the natural frequency order which we found resulted in decent fragment scores mitigating over-fragmentation. We speculate there can be more nuanced ways to carefully select this 10K subset, and leave this as a potential future work to explore.

## 8 Ethics Statement and Broader Impact

The LLMs considered in this study, Llama and Qwen family, are general purpose LLMs. Although our techniques are showing promising improvements, they are in no way ready for a production ready deployment before ensuring proper saftery checks and balances. There still needs to be more dedicated research to investigate hallucination, correctness, and completeness of the response in real-world open-ended generation.

## Acknowledgments

We thank the Ministry of Education, Govt of India, for supporting Gunjan Balde with Prime Minister Research Fellowship during his Ph.D. tenure. This research was partially funded by a Google Academic Research Award. We acknowledge National Supercomputing Mission (NSM) for providing computing resources of 'PARAM Shakti' at IIT Kharagpur, implemented by C-DAC and supported by the Ministry of Electronics and Information Technology (MeitY) and Department of Science and Technology (DST), Government of India.

## References

Marah Abdin, Jyoti Aneja, et al. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Orevaoghene Ahia, Sachin Kumar, et al. 2023. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.

Gunjan Balde, Soumyadeep Roy, et al. 2024. MEDVOC: Vocabulary adaptation for fine-tuning pre-trained language models on medical text summarization. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 6180–6188. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Gunjan Balde, Soumyadeep Roy, et al. 2025. Evaluation of LLMs in medical text summarization: The role of vocabulary adaptation in high OOV settings. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 22989–23004, Vienna, Austria. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Asma Ben Abacha and Dina Demner-Fushman. 2019. On the summarization of consumer health questions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2228–2234, Florence, Italy. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zeming Chen, Alejandro Hernández Cano, et al. 2023. Meditron-70b: Scaling medical pretraining for large language models. *Preprint*, arXiv:2311.16079.

Emma Croxford, Yanjun Gao, et al. 2025. Automating evaluation of ai text generation in healthcare with a large language model (llm)-as-a-judge. *medRxiv*, pages 2025–04.

Jiaxi Cui, Munan Ning, et al. 2024. Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model. *Preprint*, arXiv:2306.16092.

Alexander R. Fabbri, Wojciech Kryściński, et al. 2021. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Pengju Gao, Tomohiro Yamasaki, and Kazunori Imoto. 2024. Ve-kd: Vocabulary-expansion knowledge-distillation for training smaller domain-specific language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15046–15059.

Yu Gu, Robert Tinn, et al. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Suchin Gururangan, Ana Marasović, et al. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Valentin Hofmann, Hinrich Schuetze, and Janet Pierrehumbert. 2022. An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–393, Dublin, Ireland. Association for Computational Linguistics.

Jimin Hong, TaeHee Kim, et al. 2021. AVocaDo: Strategy for adapting vocabulary to downstream domain. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4692–4700, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Edward J Hu, yelong shen, et al. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Abhinav Joshi, Shounak Paul, et al. 2024. IL-TUR: Benchmark for Indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.

Guy Kaplan, Matanel Oren, et al. 2025. From tokens to words: On the inner lexicon of LLMs. In *The Thirteenth International Conference on Learning Representations*.

Zhenglun Kong, Yize Li, et al. 2025. Token reduction should go beyond efficiency in generative models–from vision, language to multimodality. *arXiv preprint arXiv:2505.18227*.

Yanis Labrak, Adrien Bazoge, et al. 2024. BioMistral: A collection of open-source pretrained large language models for medical domains. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5848–5864, Bangkok, Thailand. Association for Computational Linguistics.

Sander Land and Max Bartolo. 2024. Fishing for magikarp: Automatically detecting under-trained tokens in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11631–11646, Miami, Florida, USA. Association for Computational Linguistics.

Jinhyuk Lee, Wonjin Yoon, et al. 2020. Biobert: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Salvador Lima López, Miguel Rodríguez Ortega, et al. 2025. MultiClinSum dataset: Summarization of clinical case reports in english, spanish, french and portuguese.

Chengyuan Liu, Shihang Wang, et al. 2024. Gold panning in vocabulary: An adaptive method for vocabulary expansion of domain-specific LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7442–7459, Miami, Florida, USA. Association for Computational Linguistics.

Siyang Liu, Naihao Deng, et al. 2023. Task-adaptive tokenization: Enhancing long-form text generation efficacy in mental health and beyond. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15264–15281, Singapore. Association for Computational Linguistics.

Xing Han Lù. 2024. Bm25s: Orders of magnitude faster lexical search via eager sparse scoring. *Preprint*, arXiv:2407.03618.

Diego Mollá and Maria Elena Santiago-Martinez. 2011. Development of a corpus for evidence based medicine summarisation. In *Proceedings of the Australasian Language Technology Association Workshop 2011*, pages 86–94. Australian Language Technology Association.

Itay Nakash, Nitay Calderon, et al. 2025. Adaptivocab: Enhancing LLM efficiency in focused domains through lightweight vocabulary adaptation. In *Second Conference on Language Modeling*.

Enes Özeren, Yihong Liu, and Hinrich Schuetze. 2025. HYPEROFA: Expanding LLM vocabulary to new languages via hypernetwork-based embedding initialization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 79–96, Vienna, Austria. Association for Computational Linguistics.

Shounak Paul, Arpan Mandal, et al. 2023. Pre-trained language models for the legal domain: A case study on indian law. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, ICAIL '23, page 187–196, New York, NY, USA. Association for Computing Machinery.

Aleksandar Petrov, Emanuele La Malfa, et al. 2023. Language model tokenizers introduce unfairness between languages. In *Advances in Neural Information Processing Systems*, volume 36, pages 36963–36990. Curran Associates, Inc.

Taido Purason, Pavel Chizhov, et al. 2026. Teaching old tokenizers new words: Efficient tokenizer adaptation for pretrained models. In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 6492–6516, Rabat, Morocco. Association for Computational Linguistics.

Qwen:, An Yang, et al. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Yuval Reif, Guy Kaplan, and Roy Schwartz. 2025. Vocab diet: Reshaping the vocabulary of llms with vector arithmetic. *Preprint*, arXiv:2510.17001.

Phillip Rust, Jonas Pfeiffer, et al. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Vin Sachidananda, Jason Kessler, and Yi-An Lai. 2021. Efficient domain adaptation of language models via adaptive tokenization. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 155–165, Virtual. Association for Computational Linguistics.

Andrew Sellergren, Sahar Kazemzadeh, et al. 2025. Medgemma technical report. *Preprint*, arXiv:2507.05201.

Abhay Shukla, Paheli Bhattacharya, et al. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.

Yuqi Si, Jingqi Wang, et al. 2019. Enhancing clinical concept extraction with contextual embeddings. *Journal of the American Medical Informatics Association*, 26(11):1297–1304.

Gemma Team, Aishwarya Kamath, et al. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Hugo Touvron, Louis Martin, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Dave Van Veen, Cara Van Uden, et al. 2024. Adapted large language models can outperform medical experts in clinical text summarization. *Nature medicine*, 30(4):1134–1142.

Chaoyi Wu, Weixiong Lin, et al. 2024. Pmc-llama: toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, 31(9):1833–1843.

Atuki Yamaguchi, Aline Villavicencio, and Nikolaos Aletras. 2024. How can we effectively expand the vocabulary of llms with 0.01 gb of target language text? *arXiv preprint arXiv:2406.11477*.

Nan Zhang, Yusen Zhang, et al. 2023. Famesumm: Investigating and improving faithfulness of medical summarization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10915–10931.

Tianyi Zhang, Varsha Kishore, et al. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

# A Experiments and Results Details

## A.1 Sampling ICL Demonstration

In order to sample the ICL demonstration from train set per test example, we use cosine similarity over embeddings obtained from sentence-transformers model variant of PubMedBERT [4] for medical domain. Due to extremely lengthy nature of documents in legal domain, we use standard bm25 model inspired from prior art (Joshi et al., 2024) to get the closest training demonstration for the test point. We use bm25s library (Lù, 2024) to setup the retriever.

## A.2 Baseline Models

Here we describe the methods evaluated in this work:

**xx-BASE.** These are the BASE LLMs variant (not the instruction tuned ones): Llama-3.1-8B-BASE and Qwen2.5-7B-BASE. They have not undergone any vocabulary modification and continual pretraining.

**CPTOnly (No Vocab Adapt).** These are the variants of BASE LLMs models have not undergone any vocabulary modification but only standard continual pretraining over the domain-specific corpora.

**VocabAdapt W/o Replace.** These are pure vocabulary expansion baselines without any replacement. We directly take non-overlapping (from BASE LLMs vocabulary) top-10K vocabulary tokens learn from the domain-specififc tokenizers and add it to the model vocabulary. The expansion and

[4] https://huggingface.co/pritamdeka/PubMedBERT-mnli-snli-scinli-scitail-mednli-stsb

| | Corpus Size | SD Token Count Llama | SD Token Count Qwen | RS Token Count Llama | RS Token Count Qwen | SD OOV Conc. Llama | SD OOV Conc. Qwen | RS OOV Conc. Llama | RS OOV Conc. Qwen |
|---|---|---|---|---|---|---|---|---|---|
| **Evidence-Based Summarization** | | | | | | | | | |
| **Random** | 185 | 392 | 410 | 92 | 97 | 7.81 | 7.81 | 8.38 | 8.38 |
| ***OOV_SD*** | 185 | 346 | 354 | 75 | 79 | 14.77 | 14.77 | 10.76 | 10.76 |
| ***OOV_RS*** | 185 | 382 | 395 | 53 | 54 | 9.93 | 9.93 | 19.10 | 19.10 |
| **Clinical Healthcare Query Summarization** | | | | | | | | | |
| **Random** | 150 | 79 | 80 | 13 | 13 | 8.32 | 8.33 | 11.07 | 11.07 |
| ***OOV_SD*** | 114 | 51 | 52 | 14 | 14 | 23.30 | 23.33 | 15.14 | 15.14 |
| ***OOV_RS*** | 143 | 80 | 80 | 15 | 15 | 12.11 | 12.12 | 27.74 | 27.74 |

Table 7: Dataset statistics of summarization datasets under ***Random***, ***OOV_RS***, and ***OOV_SD*** settings, reporting mean token counts, OOV concentration (fraction of unigrams in text split more than once).

addition procedure is similar to MEDVOC (Balde et al., 2025), where for each vocabulary token to be added we iteratively add its subwords as obtained from the BASE LLM tokenizer.

**VOCABADAPT W/ Replace.** This is the replacement variant of VOCABADAPT *W/o Replace*. Here, we first replace tokens from the candidate replace set–$V_{\text{cand}}$(Eq. 2), then expand the BASE LLMs vocabulary with the remaining vocabulary tokens.

**VOCABADAPT$_{\text{Refine}}$ W/o Replace.** These are vocabulary expansion baselines without any replacement. Here, before selecting the top 10K tokens for expansion, we do a refinement steps of removing non-standard tokens–mixture of chars and numbers, chars and punctuation, numbers and punctuations–that might be inconsistent with BASE LLM tokenization, inspired from MEDVOC-LLM (Balde et al., 2025). Post-refinement we take non-overlapping (from BASE LLMs vocabulary) top-10K vocabulary tokens learn from the domain-specififc tokenizers and add it to the model vocabulary.

**VOCABADAPT$_{\text{Refine}}$ W/ Replace.** This is the replace-then-expand variant of VOCABADAPT$_{\textit{Refine}}$ *W/o Replace.*

### A.3 Results Trends

We report the full results in Table 9. We now provide discussions as observed across challenging scenarios.

**EBM and CHQ results.** The dataset statistics for these datasets is reported in Table 7. The prompt structure used for inference is reported in Table 8. We conduct evaluation in line with our challenge-oriented evaluation focusing on points that are difficult to generate (***OOV_RS***) and difficult to encode (***OOV_SD***). We note that VOCABADAPT outperforms BASE in a total of 11 out of 16 comparisons and CPTOnly in 9 out of 16 comparisons of

| **Prompt structure** |
|---|
| **Evidence-Based Summarization** |
| You are an expert medical professional.<br>####<br>Summarize the given source document in the context of the input query in 100 words or less. Use the examples to guide word choice.<br>Query 1: {Train-Query}<br>Source Document 1: {Train-Source-Document}<br>Query-Focused Summary 1: {Train-Target-Summary}<br>##<br>Query 2: {Test-Query}<br>Source Document 2: {Test-Source-Document}<br>Query-Focused Summary 2: |
| **Clinical Healthcare Query Summarization** |
| You are an expert medical professional.<br>####<br>Summarize the given patient healthcare query into a concise single question of 10 words or less. Use the examples to guide word choice.<br>Patient Health Query 1: {Train-Patient-Query}<br>Summarized Question 1 : {Train-Summarized-Question}<br>##<br>Patient Health Query 2: {Test-Patient-Query}<br>Discharge Summary 2 : |

Table 8: The prompt structure used for prompting LLMs inspired from the prompt structure proposed in Clin-Summ (Van Veen et al., 2024). Since we are using BASE LLMs there is no explicit segregation of system prompt and user prompt.

difficult scenarios around ***OOV_RS*** and ***OOV_SD*** showing exactly the same behavior as observed in the paper. This further strengthens the generalizability of our approach and supports a broader task coverage.

***OOV_SD* (High OOV in Source Documents).** In the ***OOV_SD*** setting, the test set is explicitly constructed from documents whose *inputs* exhibit the highest OOV concentration, making accurate content understanding and alignment particularly challenging. Results show that BASE and CPTOnly models degrade noticeably in both R-LCS and BERTScore, indicating that continual pretraining alone is insufficient when the source text itself is dominated by unseen or poorly tokenized terms. Vocabulary adaptation methods consistently improve performance, demonstrating that better lexical coverage at the input level directly enhances content selection and factual grounding in summaries. Refinement-based expansion yields the most stable gains, suggesting that removing noisy or irregular candidate tokens before expansion helps the model form cleaner input representations. Replacement-based variants offer additional improvements in some cases, but the gains are less uniform, highlighting the sensitivity of source-side comprehension to overly aggressive vocabulary restructuring. Overall, ***OOV_SD*** emphasizes the importance of robust input tokenization, where accurate segmentation of domain-specific terms is critical for downstream summarization quality.

| | Vocab Size | Param Incr. | Best Ckpt. | *Random* | | | | *OOV_SD* | | | | *OOV_RS* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | $FrSr_{SD}$ | $FrSr_{RS}$ | R-LCS | BSr | $FrSr_{SD}$ | $FrSr_{RS}$ | R-LCS | BSr | $FrSr_{SD}$ | $FrSr_{RS}$ | R-LCS | BSr |
| **MEDICAL –ClinSumm** | | | | | | | | | | | | | | | |
| Llama-3.1-8B-BASE | 128256 | - | - | 1.16 | 1.16 | 23.03 | 70.66 | 1.25 | 1.27 | 24.39 | 71.35 | 1.20 | 1.34 | 21.33 | 70.41 |
| CPTOnly (No Vocab Adapt) | 128256 | - | 7500 | 1.16 | 1.16 | 24.89 | 75.55 | 1.25 | 1.25 | 26.29 | 76.22 | 1.20 | 1.34 | **23.92** | 75.43 |
| VocabAdapt W/o Replace. | 141779 | 110M | 3500 | 1.05 | 1.06 | 24.81 | 75.81 | 1.09 | 1.09 | 25.89 | 76.36 | 1.06 | 1.12 | 23.41 | 75.48 |
| VocabAdapt W/ Replace. | 140251 | 98M | 3500 | 1.05 | 1.06 | 24.13 | 75.36 | 1.09 | 1.09 | 25.45 | 75.83 | 1.06 | 1.12 | 22.81 | 75.02 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 141791 | 110M | 3500 | 1.05 | 1.07 | 24.58 | 75.66 | 1.09 | 1.09 | 26.56 | 76.52 | 1.06 | 1.12 | 23.86 | 75.54 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 140263 | 98M | 3500 | 1.05 | 1.07 | **24.98** | **75.98** | 1.09 | 1.09 | **26.68** | **76.55** | 1.06 | 1.12 | 23.65 | **75.58** |
| Qwen2.5-7B-BASE | 151665 | - | - | 1.19 | 1.23 | 15.51 | 43.35 | 1.28 | 1.29 | 14.15 | 37.44 | 1.24 | 1.36 | 12.23 | 35.22 |
| CPTOnly (No Vocab Adapt) | 151665 | - | 8000 | 1.19 | 1.23 | 24.73 | 75.32 | 1.28 | 1.29 | 25.96 | 75.90 | 1.24 | 1.36 | 22.41 | 74.44 |
| VocabAdapt W/o Replace. | 162738 | 79M | 3500 | 1.09 | 1.10 | **25.04** | **75.72** | 1.12 | 1.11 | 26.11 | 76.15 | 1.10 | 1.14 | **23.12** | **75.21** |
| VocabAdapt W/ Replace. | 158751 | 50M | 3500 | 1.09 | 1.10 | 24.67 | 75.41 | 1.12 | 1.11 | **26.43** | **76.19** | 1.10 | 1.14 | 23.06 | 74.98 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 162745 | 79M | 3500 | 1.08 | 1.09 | 24.22 | 75.26 | 1.12 | 1.11 | 25.76 | 75.84 | 1.10 | 1.14 | 22.40 | 74.77 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 158758 | 51M | 3500 | 1.08 | 1.09 | 24.68 | 75.51 | 1.12 | 1.11 | 26.21 | 76.12 | 1.10 | 1.14 | 22.81 | 74.93 |
| **MEDICAL –EBM** | | | | | | | | | | | | | | | |
| Llama-3.1-8B-BASE | 128256 | - | - | 1.07 | 1.08 | 18.31 | 67.14 | 1.24 | 1.12 | 17.75 | **71.66** | 1.12 | 1.30 | 15.90 | 68.20 |
| CPTOnly (No Vocab Adapt) | 128256 | 0M | 7500 | 1.07 | 1.08 | 19.99 | 74.90 | 1.24 | 1.12 | **17.76** | 71.30 | 1.12 | 1.30 | **16.58** | **72.56** |
| VocabAdapt W/o Replace. | 141779 | 110M | 3500 | 1.01 | 1.01 | 20.18 | 72.40 | 1.09 | 1.02 | 17.12 | 69.10 | 1.03 | 1.12 | 16.44 | 72.21 |
| VocabAdapt W/ Replace. | 140251 | 98M | 3500 | 1.01 | 1.01 | 20.20 | 73.99 | 1.09 | 1.02 | 16.85 | 69.73 | 1.03 | 1.12 | 16.30 | 72.23 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 141791 | 110M | 3500 | 1.01 | 1.01 | **20.60** | 72.38 | 1.09 | 1.02 | 16.85 | 70.45 | 1.03 | 1.12 | 16.34 | 72.44 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 140263 | 98M | 3500 | 1.01 | 1.01 | 19.43 | **74.28** | 1.09 | 1.02 | 16.74 | 71.05 | 1.03 | 1.12 | 16.24 | 72.23 |
| Qwen2.5-7B | 151665 | - | - | 1.13 | 1.13 | 15.31 | 55.33 | 1.26 | 1.17 | 14.53 | 62.12 | 1.17 | 1.33 | 13.29 | 55.76 |
| CPTOnly (No Vocab Adapt) | 151665 | 0M | 8000 | 1.13 | 1.13 | 17.79 | 63.70 | 1.26 | 1.17 | 14.39 | 57.32 | 1.17 | 1.33 | 12.27 | 55.93 |
| VocabAdapt W/o Replace. | 162738 | 79M | 3500 | 1.05 | 1.05 | **20.06** | 73.13 | 1.12 | 1.07 | **16.58** | **71.44** | 1.07 | 1.14 | 15.08 | 70.10 |
| VocabAdapt W/ Replace. | 158751 | 50M | 3500 | 1.05 | 1.05 | 19.77 | **73.90** | 1.12 | 1.07 | 16.56 | 71.43 | 1.07 | 1.14 | **15.19** | **70.81** |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 162745 | 79M | 3500 | 1.05 | 1.05 | 19.48 | 70.15 | 1.12 | 1.07 | 15.40 | 64.45 | 1.07 | 1.14 | 14.80 | 63.72 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 158758 | 51M | 3500 | 1.05 | 1.05 | 18.99 | 69.80 | 1.12 | 1.07 | 15.41 | 66.21 | 1.07 | 1.14 | 14.78 | 66.32 |
| **MEDICAL –CHQ** | | | | | | | | | | | | | | | |
| Llama-3.1-8B-BASE | 128256 | - | - | 1.08 | 1.17 | **43.81** | 83.44 | 1.35 | 1.25 | 50.06 | 85.11 | 1.39 | 1.51 | **48.54** | 84.23 |
| CPTOnly (No Vocab Adapt) | 128256 | 0M | 7500 | 1.08 | 1.17 | 43.30 | 83.67 | 1.35 | 1.25 | 51.32 | 85.94 | 1.39 | 1.51 | 47.69 | **84.93** |
| VocabAdapt W/o Replace. | 141779 | 110M | 3500 | 1.06 | 1.08 | 43.69 | **83.98** | 1.28 | 1.13 | 50.37 | **86.03** | 1.09 | 1.27 | 46.88 | 84.52 |
| VocabAdapt W/ Replace. | 140251 | 98M | 3500 | 1.06 | 1.08 | 42.99 | 83.81 | 1.28 | 1.13 | **51.41** | 85.94 | 1.09 | 1.27 | 46.67 | 84.38 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 141791 | 110M | 3500 | 1.06 | 1.08 | 42.14 | 83.51 | 1.28 | 1.13 | 50.22 | 85.28 | 1.09 | 1.27 | 46.58 | 84.62 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 140263 | 98M | 3500 | 1.06 | 1.08 | 41.58 | 83.31 | 1.28 | 1.13 | 50.60 | 85.69 | 1.09 | 1.27 | 46.82 | 84.70 |
| Qwen2.5-7B | 151665 | - | - | 1.09 | 1.17 | 40.59 | 83.84 | 1.36 | 1.25 | 46.56 | 85.17 | 1.52 | 1.51 | 49.60 | 85.27 |
| CPTOnly (No Vocab Adapt) | 151665 | 0M | 8000 | 1.09 | 1.17 | **43.09** | **84.02** | 1.36 | 1.25 | 51.44 | 85.99 | 1.52 | 1.51 | 48.67 | 84.75 |
| VocabAdapt W/o Replace. | 162738 | 79M | 3500 | 1.07 | 1.08 | 42.09 | 83.55 | 1.29 | 1.13 | 49.22 | 85.90 | 1.09 | 1.27 | 48.08 | 84.99 |
| VocabAdapt W/ Replace. | 158751 | 50M | 3500 | 1.07 | 1.08 | 41.28 | 83.24 | 1.29 | 1.13 | 50.00 | 86.20 | 1.09 | 1.27 | 48.25 | 85.12 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 162745 | 79M | 3500 | 1.07 | 1.08 | 41.60 | 83.77 | 1.29 | 1.13 | 47.86 | 85.74 | 1.09 | 1.27 | 47.26 | 85.10 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 158758 | 51M | 3500 | 1.07 | 1.08 | 42.85 | 83.83 | 1.29 | 1.13 | 47.41 | 85.15 | 1.09 | 1.27 | 48.22 | 85.19 |
| **LEGAL** | | | | | | | | | | | | | | | |
| Llama-3.1-8B-BASE | 128256 | - | - | 1.03 | 1.03 | **25.89** | 67.04 | 1.08 | 1.06 | 24.82 | 64.10 | 1.06 | 1.08 | **24.86** | 65.28 |
| CPTOnly (No Vocab Adapt) | 128256 | - | 10000 | 1.03 | 1.03 | 25.36 | 69.05 | 1.08 | 1.06 | 24.83 | 68.04 | 1.06 | 1.08 | 24.36 | **68.38** |
| VocabAdapt W/o Replace. | 139470 | 91M | 6500 | 1.01 | 1.01 | 25.47 | 69.05 | 1.01 | 1.01 | 24.17 | 67.45 | 1.01 | 1.01 | 23.71 | 67.64 |
| VocabAdapt W/ Replace. | 137942 | 79M | 6500 | 1.01 | 1.01 | 25.42 | **69.14** | 1.01 | 1.01 | **24.89** | **68.12** | 1.01 | 1.01 | 23.92 | 68.11 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 139653 | 93M | 6500 | 1.01 | 1.01 | 24.91 | 68.86 | 1.01 | 1.01 | 24.49 | 67.72 | 1.01 | 1.01 | 23.59 | 67.71 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 138125 | 81M | 6500 | 1.01 | 1.01 | 24.61 | 68.59 | 1.01 | 1.01 | 23.93 | 67.28 | 1.01 | 1.01 | 23.41 | 67.55 |
| Qwen2.5-7B-BASE | 151665 | - | - | 1.06 | 1.06 | 10.63 | 28.25 | 1.11 | 1.10 | 9.97 | 27.64 | 1.09 | 1.12 | 9.90 | 27.15 |
| CPTOnly (No Vocab Adapt) | 151665 | - | 10500 | 1.06 | 1.06 | **25.68** | **69.04** | 1.11 | 1.10 | **25.16** | **67.60** | 1.09 | 1.12 | **24.72** | **67.97** |
| VocabAdapt W/o Replace. | 162206 | 76M | 6500 | 1.02 | 1.02 | 23.31 | 66.83 | 1.05 | 1.05 | 22.22 | 65.25 | 1.04 | 1.05 | 21.95 | 65.67 |
| VocabAdapt W/ Replace. | 158219 | 47M | 6500 | 1.02 | 1.02 | 24.12 | 67.89 | 1.05 | 1.05 | 23.32 | 66.25 | 1.04 | 1.05 | 22.91 | 66.94 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/o Replace. | 162352 | 77M | 6500 | 1.02 | 1.02 | 23.54 | 67.29 | 1.05 | 1.05 | 23.08 | 66.04 | 1.04 | 1.05 | 22.26 | 66.13 |
| $\text{VocabAdapt}_{\text{Refine}}$ W/ Replace. | 158365 | 48M | 6500 | 1.02 | 1.02 | 24.23 | 67.89 | 1.05 | 1.05 | 23.60 | 66.19 | 1.05 | 1.05 | 23.22 | 66.89 |

Table 9: Comparison of vocabulary adaptation methods across different Legal and Medical domains using in-context learning with 1 ICL demonstration in two challenging scenarios–***OOV_RS*** and ***OOV_SD***, and a ***Random*** subset. Performance is measured using Rouge-LCS (R-LCS) and BertScore (BSr) metrics.

***OOV_RS* (High OOV in Reference Summaries).** In contrast, ***OOV_RS*** focuses on datapoints where the *references*—rather than the sources—contain high OOV concentration, stressing the model's ability to generate or align with rare and domain-specific lexical forms. While BASE and CPTOnly models perform reasonably on the ***Random*** split, they lag behind vocabulary-adapted models in ***OOV_RS***, particularly in R-LCS, indicating difficulty in matching reference phrasing and terminology. Vocabulary expansion significantly narrows this gap, with consistent improvements across backbones, confirming that enhanced output-side lexical expressivity enables closer overlap with reference summaries. Refinement again proves beneficial by stabilizing gains across both metrics, whereas replacement-based methods yield modest but less consistent improvements. The contrast between ***OOV_SD*** and ***OOV_RS*** highlights that source-side OOV primarily affects content understanding, while reference-side OOV impacts lexical realization, and effective vocabulary adaptation is crucial in addressing both challenges beyond what is observed in the ***Random*** setting.

**Implications.** Taken together, these results demonstrate that the benefits of vocabulary expansion scale with the severity and location of vocabulary mismatch. When OOVs are heavily concentrated in reference summaries (***OOV_RS***), vocabulary expansion directly improves generation fidelity. When OOVs originates in the source document (***OOV_SD***), vocabulary expansion becomes critical, yielding the largest and most consistent improvements. These findings highlight vocabulary mismatch as a bottleneck in expert domain adaptation and suggest that it should be selectively applied based on domain characteristics where vocabulary mismatch is indeed a significant problem.

### A.4 LLM-as-a-Judge Evaluation

We conduct LLM-as-a-judge (LlaJ) evaluation for summaries generated in medical and legal domain. We gather a uniform random subset of 100 summaries for medical domain and 20 summaries for legal domain from ***OOV_SD*** and ***OOV_RS*** settings distributed uniformly accorss Llama and Qwen models. We use models from Google's Gemma3 family (Team et al., 2025): MedGemma-27B-text-it[5] model as our judge model for medical domain,

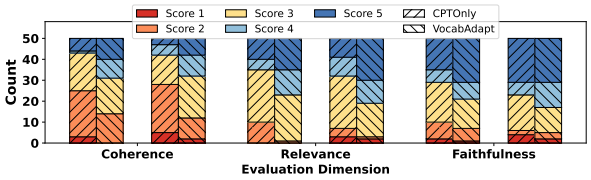



Figure 2: We report the score distribution as obtained from our LLM-as-a-Judge evaluation for medical domain. Acoss each dimension, the bars to the left corresponds to Llama model and bars to the right are for Qwen model. We note consistently VOCABADAPT results in higher score of 4 or 5 compared to CPTOnly.

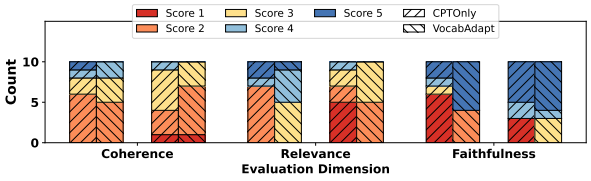



Figure 3: We report the score distribution as obtained from our LLM-as-a-Judge evaluation for Legal domain. Acoss each dimension, the bars to the left corresponds to Llama model and bars to the right are for Qwen model.

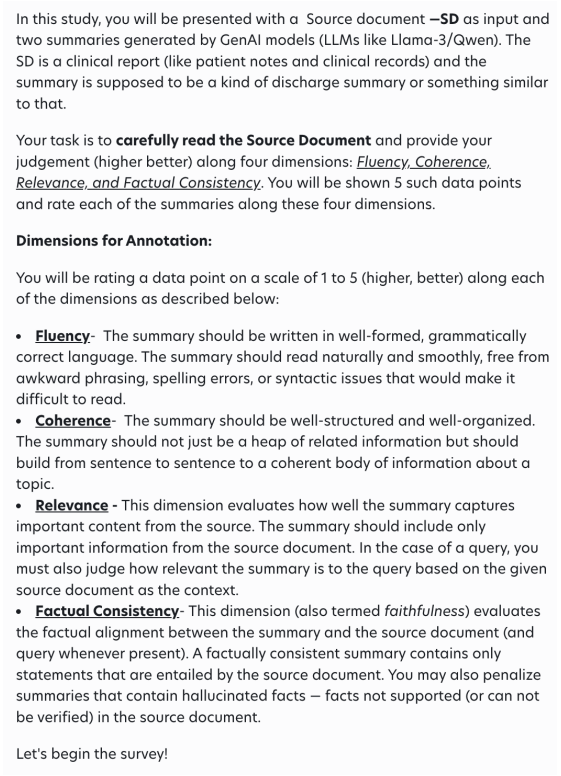
and Gemma3-27B-it[6] as our judge model for Legal domain. The model is provided as input the source document and a generated summary. It is then asked to rate the generated summary on three dimensions in three separate runs: (i) coherence, (ii) relevance, and (iii) faithfulness; each on a scale of 1-5 (higher better). This is done for both VOCABADAPT and CPTOnly summaries. The detailed system prompts and user prompts for each of the settings are available in the codebase inside the folder "Random-Eval-LlaJ" folder. Our final scores are reported as average across across summary pairs in Table 4 and Figures 2 and 3.

### A.5 Supplementary Human Evaluation

We conducted an additional human assessment across fluency, consistency, relevance, and coherence rated on a scale of 1-5 on a subset of 20 summaries for the medical domain. The annotation instructions are shown in Figure 4. The evaluation was conducted on the Prolific platform with the

[5] https://huggingface.co/google/medgemma-27b-text-it

[6] https://huggingface.co/google/gemma-3-27b-it



In this study, you will be presented with a Source document **–SD** as input and two summaries generated by GenAI models (LLMs like Llama-3/Qwen). The SD is a clinical report (like patient notes and clinical records) and the summary is supposed to be a kind of discharge summary or something similar to that.

Your task is to **carefully read the Source Document** and provide your judgement (higher better) along four dimensions: *Fluency, Coherence, Relevance, and Factual Consistency*. You will be shown 5 such data points and rate each of the summaries along these four dimensions.

**Dimensions for Annotation:**

You will be rating a data point on a scale of 1 to 5 (higher, better) along each of the dimensions as described below:

- **Fluency**- The summary should be written in well-formed, grammatically correct language. The summary should read naturally and smoothly, free from awkward phrasing, spelling errors, or syntactic issues that would make it difficult to read.
- **Coherence**- The summary should be well-structured and well-organized. The summary should not just be a heap of related information but should build from sentence to sentence to a coherent body of information about a topic.
- **Relevance** - This dimension evaluates how well the summary captures important content from the source. The summary should include only important information from the source document. In the case of a query, you must also judge how relevant the summary is to the query based on the given source document as the context.
- **Factual Consistency**- This dimension (also termed *faithfulness*) evaluates the factual alignment between the summary and the source document (and query whenever present). A factually consistent summary contains only statements that are entailed by the source document. You may also penalize summaries that contain hallucinated facts – facts not supported (or can not be verified) in the source document.

Let's begin the survey!

Figure 4: Annotation instructions shown to the participants.

following participation criterion:

- ***Highest education level completed***. Undergraduate degree (BA/BS/other) *OR* Graduate degree (MA/MS/MPhil/other) *OR* Doctorate degree (PhD/other)
- **Employment Status.** Full-Time *OR* Part-Time *OR* Due to start a new job within the next month
- **Subject**: Biochemistry (Molecular and Cellular) *OR* Biological Sciences *OR* Biology *OR* Biomedical Sciences

Each annotator was shown five summary pairs and each summary pair was evaluated independently by three annotators. The median time to complete the study was 20 minutes. In total 12 annotators were hired for the evaluation task. All the annotators were compensated at a rate of GBP 9 per hour. The average results across each category of annotation are shown in Table 10.

The human evaluation exhibits the same overall trend as the LLM-as-a-Judge results; vocabulary adaptation models generate better summaries than the CPTOnly counterpart; reinforcing the validity

| Model | Fluency | Coherence | Relevance | Factual Consistency | Overall |
|---|---|---|---|---|---|
| CPTOnly | 4.12 | 4.23 | 4.17 | 4.25 | 4.19 |
| VOCABADAPT | **4.35** | **4.40** | **4.32** | **4.57** | **4.41** |

Table 10: Human evaluation trends comparing competing baseline and our propose VOCABADAPT method.

of our conclusions. However, we must here mention conducting domain-specific human evaluation presented substantial practical challenges. Evaluating only 20 summary pairs in the medical domain incurred us a cost of approximately **GBP 48 via Prolific** (GBP 36 for annotators and GBP 12 as platform fees). ***In the legal domain, Prolific does not even provide a sufficiently large or appropriate participant pool to enable reliable evaluation***. These constraints make large-scale domain-expert evaluation difficult to sustain.

We therefore view LLM-as-a-Judge not as a replacement for human evaluation, but as a scalable and reproducible alternative that is particularly valuable in settings where domain expertise is limited, costly, or difficult to source. Our results demonstrate that, when validated against human judgments, it provides consistent and reliable comparative assessment.