

ProveTok: Proof-Carrying Budgeted Evidence Tokenization for Grounded 3D Report Generation

ProveTok: Proof-Carrying Budgeted Evidence Tokenization for Grounded 3D Report Generation

应用：3D CT Radiology Report Generation with Pixel-level Grounding

1. 一句话 Punchline (Oral gatekeeper 版)

在严格预算 B 下, ProveTok 生成带显式 3D 支持域 Ω 的证据 tokens, 并以 **proof-carrying** 生成协议强制每条断言携带可机检证据与可审计 verifier trace; 当证据不足时触发可约束的 **refusal calibration**; 跨多预算呈现可解释的 **scaling** 与 Pareto, 并用 pixel-level **grounding** 与 **counterfactual** 实验证明 citations 不是装饰。

Reviewer-check (硬性)

- 句子里必须出现：预算 B 、 Ω 、**proof-carrying**、**refusal calibration**、**scaling**、**grounding**、**counterfactual** ✓
-

2. 背景与动机 (仅 4 段, 段段有用)

2.1 预算失衡：3D→text 的 compute 不可控

3D 体数据包含大量冗余体素；fixed-grid 全量 tokenization 在 3D 上计算爆炸，而 2D/2.5D slice/ROI crop 虽省算力却丢失 3D 证据并引入启发式偏差。我们关心的是：在严格预算 B 下，模型应如何把 compute 分配到最“证据密集”的空间区域与推理步骤。本文将 B 定义为联合预算

```
[  
B=B_{\text{enc}}+B_{\text{gen}},  
]
```

分别约束证据 tokenization 的 encoder compute 与报告生成的 decoder compute；所有对比在 **FLOPs-matched** 或 **latency-matched** 协议下报告（把“省算力”从口号变成可检验约束）。

Reviewer-check

- 这段必须让 B 成为主轴 ✓
 - 明确指出 fixed-grid vs slice/ROI 的失衡 ✓
-

2.2 可信失衡：报告“像”不等于“有证据”

在临床文本生成中，hallucination 成本极高；传统做法把 groundedness 当作事后评测或人工抽检，无法把“证据不足时应拒答/不确定”变成可控机制。近期口径是把 trustworthiness 变成硬指标与训练目标：例如在检索增强生成中同时评估 citation groundedness 与“该拒答时拒答”的能力，并把拒答质量当作可校准信号（reliability/ECE），同时用漏报率约束防止“封嘴”。这与 ICLR 2025 的 Trust-Align 类工作将 grounded attributions 与 learning-to-refuse 联合作为信任目标的趋势一致。（OpenReview）

Reviewer-check

- 必须对齐 oral 趋势，并引出“拒答可校准 + 防封嘴” ✓
-

2.3 我们的重写：Budgeted Evidence Tokenization → Proof-Carrying Generation

我们将 3D 报告生成重写为两个耦合问题：

- (1) **BET**: 在预算 $B_{\{\text{text}\}}^{\{\text{enc}\}}$ 内生成证据 tokens，每个 token 绑定显式 3D 支持域 Ω ；
- (2) **PCG**: 生成报告时，每条临床断言必须携带 token-citation，并由可复现 verifier 检查；若证据不足，则以可校准方式拒答/不确定。

关键重写在于：我们不再把 groundedness 当作事后指标，而是把它写进生成协议——**每条断言必须“携证据上岗”**，否则 verifier 必须能指明失败类型并触发 refine 或拒答，从而把“像真的”与“有证据”分离。

Reviewer-check

- 必须明确“重写问题形式”，否则会被喷系统工程 ✓
-

2.4 我们证明什么（三类主结果）

1. **Efficiency-Correctness**: 在多预算 B 上呈现 Pareto dominate，并拟合可解释 scaling law 与 compute allocation model（对齐 test-time compute / inference scaling 口径）。我们借鉴 OpenReview 上关于“在计算约束下预测最优推理配置与分配”的 RAG inference scaling 叙事：不仅画曲线，还要能预测最优分配并给出 regret。（OpenReview）
2. **Grounding**: 在具有 pixel-level 3D segmentation 的数据上，citation-grounding (IoU/Dice/hit-rate) 显著更强。（arXiv）

3. **Non-triviality**: Permutation / citation-swap / evidence-drop 等反事实实验显著击穿，证明 citations 不是 attention 装饰，而是可验证的因果依赖。

Reviewer-check

- 这段等价于 oral 的“主图导读”：Pareto+scaling、grounding、counterfactual 三板斧 ✓
-

3. 问题定义（用 B 统治全篇）

给定 3D 体数据 (V) 与预算 ($B=B_{\text{enc}}+B_{\text{gen}}$) (详见 § 7.3 公平性协议)，学习一个闭环系统。

3.0 离散化约定（保证 verifier 可机检）

在体素网格上定义 cell family (\mathcal{G}) (如八叉树节点或规则 block)。每个支持域

```
[  
\Omega \equiv \texttt{cell\_id}\in\mathcal{G}  
]
```

是可枚举索引；其对应体素集合由确定性函数

```
[  
\phi(\texttt{cell\_id})\rightarrow\{(x,y,z)\}  
]
```

给出，使得 verifier 不依赖学习模型即可回溯证据。

3.1 证据 tokenization (BET)

输出证据 tokens:

```
[  
T=\{(t_i,\Omega_i,s_i)\}_{i=1}^{|T|}, \quad |T|\leq B_{\text{enc}}  
]
```

其中 ($t_i \in \mathbb{R}^d$) 为 cell 特征 (3D encoder 在 ($\phi(\Omega_i)$) 上 pooling 得到)， (Ω_i) 为显式 3D 支持域 (cell_id)， ($s_i \in [0,1]$) 为 evidence head 的摘要分数 (finding relevance / uncertainty，用于 greedy (Δ) 与 refusal 决策)。

3.2 proof-carrying 生成 (PCG)

生成报告 (y)，并对每条关键断言 (frame) (k) 输出引用集合

```
[
```

$C_k \subseteq \{1, \dots, |T|\}, \quad |C_k| \leq K_{\max}$.

]

同时输出 refusal/uncertainty 标记与支持概率 ($q_k \in [0,1]$) (用于 refusal calibration)。

为防止“引用倾倒”，我们施加硬约束：每条断言必须引用不超过 (K_{\max}) 个 token，并在评测中监控 citation-dump 模式（分析项，不单独作为 taxonomy，以免被喷“发明新指标”）。

3.3 程序化验证 (Verifier)

给定 ((y,{C_k},T,\mathcal{E})) 输出 issue 列表 (taxonomy 固定、可枚举、可审计)：

- **unsupported**: 断言无可接受证据
- **overclaim**: 证据不足以支撑强断言 (粒度过细, 如 size/location 过细)
- **inconsistency**: 互相矛盾/与结构槽冲突
- **missing-slot**: 结构化必要槽缺失

每条 issue 记录 ((k,\text{issue_type},\text{severity},\text{rule-id},\text{evidence_trace})), 其中 severity $\in \{\text{critical, non-critical}\}$, evidence_trace 为最小可复现对象 (token ids、cell_id、触发规则 id)。

Reviewer-check

- Taxonomy 必须可枚举, 否则 verifier 变成黑箱 ✓
 - 这里没有引入“第三贡献”，只是把 PCG 协议写清 ✓
-

4. 方法 (只写闭环: Tokenize → Generate(+cite) → Verify → Refine)

4.1 总览 (Fig1)

输入: ((V,B))

闭环: BET 产生 tokens \rightarrow PCG 生成断言+引用 \rightarrow Verifier 输出 issues \rightarrow 若预算未耗尽则按 issues/不确定性 refine tokens

输出: 报告 + citations + verifier trace (可审计 artifact)

Fig1 应该画什么 (必须是协议闭环, 不是模块清单)

- 左: 3D volume + coarse cells
 - 中: tokens (带 $\Omega/\text{cell_id}$) \rightarrow frames+citations \rightarrow verifier issues
 - 右: refine loop (priority queue / Δ 最大 split)
 - 输出: 可审计 trace (per-step Δ 、issues 变化、citations)
-

4.2 C1: Budgeted Evidence Tokenization (BET)

4.2.1 表示与层级

用层级 partition (如八叉树) 构造 coarse-to-fine cells。初始深度 (d_0) 固定、最大深度 (d_{\max}) 固定；split 操作为标准 8-children octree split。

4.2.2 预算分配: Deterministic Greedy (主方法, 保证可复现)

我们定义每个 cell 的边际证据收益 ($\Delta(c)$)，来源于当前 verifier issues 与 evidence head 不确定性 (可解释、可计算、可复现)：

Algorithm 1: Deterministic Greedy BET-Refine

Input: (V), (B_{enc}), (d_0), (d_{\max}), stop threshold (ϵ), verifier refresh period (M)

State: cell set (S) (初始为 depth (d_0) 全覆盖) , tokens ($T(S)$), issues (I)

1. 构建/更新 evidence graph (E) (见 § 4.3.1)
 2. 对每个可 split cell ($c \in S$) ($\text{depth} < d_{\max}$) 计算
$$\begin{aligned}\Delta(c) = & \underbrace{\sum_{u \in I} w_u \cdot \widehat{\Delta}_{\text{issue}}(u, c)}_{\text{verifier-driven}} \\ & + \lambda \underbrace{H(p(\text{critical findings} | c))}_{\text{uncertainty}}\end{aligned}$$
 - (w_u) 按 severity 固定 (critical > non-critical) , ($\widehat{\Delta}_{\text{issue}}$) 用 evidence head 的局部预测近似 (避免每步跑 full verifier)
 - ($H(\cdot)$) 用 entropy/margin (完全可计算)
 3. 选择 ($c^* = \arg \max \Delta(c)$), tie-break: 最小 cell_id (字典序)
 4. 若 ($\Delta(c^*) < \epsilon$) 或 ($|S| \geq B_{\text{enc}}$): 停止
 5. split (c^*) 为 8 children, 增量更新 tokens 与 priority queue (缓存 encoder 特征, 仅对新增子 cell 编码/ pooling)
 6. 每 (M) 步运行一次 PCG+verifier 刷新 (I) (其余步用近似项)
- Output:** tokens ($T(S)$)

停机阈值 (ϵ) (避免被喷调参)

(ϵ) 在开发集用 Δ 的固定分位点规则设定 (例如候选 Δ 的 5% 分位) , 并在所有预算 B 上共享, 不按 B 单独调。

Reviewer-check

- 主文不依赖 RL 

- (\Delta) 必须可解释 (来自 issues/不确定性) ✓
- deterministic 的 tie-break 必须写死 ✓

4.2.3 Learned allocator (仅消融)

可选: contextual bandit 学 ($\widehat{\Delta}$) 的排序近似, reward = issue reduction + external grounding gain (归一化), 仅用于证明“不是启发式凑出来”, 不作为系统可用性的前提。

4.3 C2: Proof-Carrying Generation (PCG)

4.3.1 Claim space: 结构化 finding frames (避免自由文本无边界)

我们将临床断言规范为 finding frames (有限槽) :

```
[  
(\text{finding type}, \text{laterality}, \text{anatomical location}, \text{severity/size bin},  
\text{negation/uncertain})  
]
```

证据图 (\mathcal{E}) (可枚举合法域)

对每个 token (i), evidence head 输出槽值候选及其置信度:

```
[  
\mathcal{E}(i)=\{(\text{type}=nodule,p),(\text{loc}=LLL,p),(\text{lat}=left,p),  
(\text{sizebin}=3\text{--}5\text{mm},p),\dots  
]
```

全局合法域: ($\mathcal{V}_{\text{slot}} = \bigcup_i \mathcal{E}(i)$)。constrained decoding 只允许输出域内槽值。

4.3.2 双通道输出协议 (堵正文“加戏”)

生成器必须输出:

- findings table: (f_k, C_k, q_k);
- 叙述文本: 由 (a) 通过固定模板或受限重写生成, 句末附 [t_i]。

Verifier 首先检查 (b) 是否可无损回译到 (a) (有限模板+词表), 否则记 inconsistency。这样正文无法绕开 frames 私自“加戏”。

4.3.3 Token-citation: 每条断言必须输出 (C_k)

对每个 frame (f_k), 模型输出指针分布 ($p(i|f_k)$) 并选 Top-(K_{\max}) 形成 (C_k), 同时输出 coverage/support score (供 verifier 判 unsupported/overclaim)。训练时 (C_k) 可用 weak supervision: 来自 evidence head 的最高支持 token; 在有 grounding 标签处 (ReXGroundingCT) 可加 overlap 监督。(arXiv)

4.3.4 Verifier：可复现、可枚举、可审计（Taxonomy v1.0 锁死）

Verifier 只用确定性规则（rule-id 版本锁定，防止“后验改规则”）：

- **U1 unsupported**: 对 claim (k)，若 ($\forall i \in C_k, \text{support}(f_k, i) = 0$)
- **O1 overclaim (粒度)**：预定义粒度层级（location: lobe > segment > subsegment；size: bin > exact mm）。若输出细于证据图可支持层级则 O1
- **I1 inconsistency**: 互斥属性冲突（negation vs positive；left vs right；不相交 severity bins）
- **M1 missing-slot**: 按 finding type 固定 required slots，缺任一则 M1
每条 issue 同时标注 severity (critical / non-critical)，critical set 在附录枚举并固定（例如 pneumothorax / pleural effusion / large consolidation / suspicious nodule 等，按任务定义锁死）。

4.3.5 Calibrated refusal: 拒答不是逃避

每个 frame 输出支持概率 ($q_k \in [0,1]$)，表示“在给定 citations 下可被 verifier 接受”的概率。若

[

$q_k < \tau_{\text{refuse}}$

]

则输出 uncertain/refuse；否则输出具体槽值。阈值 (τ_{refuse}) 在开发集按约束 **critical miss-rate** $\leq \delta$ 选择一次，并在所有预算 B 上固定，防止“按预算调阈值刷表”。

必须同时报告（反封嘴主表约束）

- unsupported rate (越低越好)
- critical miss-rate (不得上升)
- refusal rate (拒答比例)
- refusal ECE / reliability (校准好坏)

Reviewer-check

- 这节必须让 reviewer 相信：不是“定义规则封嘴”，而是把不确定性变成可校准行为，并用漏报率约束 ✓
- claim space 必须有边界，否则 verifier 形同虚设 ✓

5. 预算 Scaling Laws 与 Compute Allocation (Fig2 主轴)

我们在多个预算 ($B \in \{B_1, \dots, B_m\}$) 下评测，并拟合性能随预算的规律，同时学习可预测的 compute allocation model。

5.1 Performance-Budget scaling (可解释拟合，不绑死形状)

对每个性能指标 (P) (correctness、grounding、trustworthiness、Vol-Trust 等) , 拟合两类函数族并用 AIC/BIC 选型:

- 饱和幂律: [

$$P(B)=P_{\infty}-a(B+b_0)^{-\alpha}$$

]

- 对数饱和: [

$$P(B)=c_0+c_1 \log(B+b_0) \quad \text{quad}(\text{并截断到 } [0,1])$$

]

拟合只在开发集进行; 测试集报告预测误差与曲线稳定性。

5.2 Allocation model (必须能“预测最优分配”，不是只画曲线)

统一 compute 单位为 FLOPs:

```
[  
\text{FLOPs}\{total\}=|\text{FLOPs}\{\text{enc}\}(B_{\text{enc}})+\text{FLOPs}\{\text{dec}\}(B_{\text{gen}})+\text{FLOPs}\{\text{verify}\}(n_{\text{verify}})  
]
```

学习一个回归预测器 ($\hat{P}(\theta)$)，其中 ($\theta = (B_{\text{enc}}, n_{\text{refine}}, B_{\text{gen}}, n_{\text{verify}})$)。通过网格采样少量配置点训练 (例如 30–50 个配置)，再解约束优化：

```
[  
\max_{\theta} \hat{P}(\theta) \quad \text{quad s.t. } \text{FLOPs}_{total}(\theta) \leq \mathcal{B}  
]
```

得到在预算 (\mathcal{B}) 下的**预测最优配置**。测试时报告 regret (预测最优 vs 实际最优差距)，以证明“模型能预测最优分配”。该叙事与 RAG inference scaling work 在 OpenReview 上提出的 computation allocation model 口径对齐。[\(OpenReview\)](#)

Fig2 应该画什么 (必须包含“可预测分配”)

- 左: 多预算 Pareto (correctness vs compute; grounding vs compute; trustworthiness vs compute)
- 中: $P(B)$ 拟合曲线 + 边际收益递减点 (stop principle)
- 右: allocation model 预测最优配置 vs 真实最优配置 (regret 曲线)

Reviewer-check

- 必须输出“模型/拟合”，不能只给曲线 ✓
- 必须能预测最优分配并报告 regret ✓

6. 数据与评测 (硬地基: 公开数据 + pixel-level grounding)

6.1 Report generation 数据 (公开可复现)

- **CT-RATE**: 公开 3D chest CT volumes paired with reports; 论文与数据卡描述其包含 25,692 non-contrast 3D chest CT scans、21,304 unique patients，并可扩展到 50,188（多重重建）。([arXiv](#))
- **CT-3DRRG**: 由 Argus 工作整理并描述为“largest publicly available 3D CT-report dataset”，用于跨源泛化与训练/评测 recipe 对比。[\(ACL Anthology\)](#)

6.2 Pixel-level grounding 数据 (硬 grounding)

- **ReXGroundingCT**: 公开数据集，将 free-text findings 与 pixel-level 3D segmentation 关联；包含 3,142 CT 与 finding-level grounding，且明确其为 3D chest CT 上的 sentence-level grounding 资源。[\(arXiv\)](#)

6.3 泄漏防护 (Protocol Lock, 必须像法律条款)

1. **ID 统一**: $(\text{\\texttt{scan_hash}} = \text{\\text{SHA256}}(\text{\\text{patient_id}} || \text{\\text{study_date}} || \text{\\text{series_uid}}))$; split 与交集仅基于 scan_hash 与 patient_id。
2. **patient-level split**: 同一 patient_id 的任何 scan 只能出现在 train/val/test 之一。
3. **去重**:
 - 文本：报告归一化后 MinHash + Jaccard > 0.9 判 near-duplicate；重复只保留 1 个并记录映射表。
 - 影像：固定下采样后 perceptual hash / 随机投影 hash 判 duplicate。
4. **交集处理 (锁死)**：禁止 ReXGroundingCT val/test 进入训练、阈值选择 $((\text{\\epsilon}, \text{\\tau}_{\text{refuse}}))$ 、allocation model 拟合。
5. **版本锁定**：数据版本 (revision hash)、split manifest (所有 scan_hash 列表) 与代码 commit 写入 artifact。

Reviewer-check

- 这节是“能否避免一票否决”的关键，必须写死

7. 实验设计 (3 图 2 表；指标与 baseline 必须齐)

7.1 主指标 (Table1)

Clinical correctness (结构化事实)

- finding frame F1 (含 laterality/location/negation/size bin 等槽)

- 匹配方式：按 finding type + coarse location + laterality 做 Hungarian matching；slot-level micro/macro F1 统计

Grounding (ReXGroundingCT) ([arXiv](#))

- sentence → citation → 3D segmentation:
 - hit-rate：是否存在被引用 token 的 (\Omega) 与 lesion mask 有 overlap (≥ 阈值)
 - IoU / Dice：对 cited cells 的 union 与 lesion mask 计算 (同时报告 max-over-citations 与 union-over-citations 两种口径)

Trustworthiness (协议性指标, verifier taxonomy)

- unsupported / overclaim / inconsistency / missing-slot rates (按 severity 分桶)
- Vol-Trust：加权组合（仅做汇总主指标，不把它当贡献）

Safety against “封嘴”

- critical finding miss-rate
- refusal calibration：ECE / reliability diagram (按 (q_k) 分桶)

Efficiency

- #tokens、encoder/decoder/verifier FLOPs (统一口径)
 - end-to-end latency：mean + P95 (cold / warm cache 分开)
-

7.2 Baselines (必须齐，否则拒)

Tokenization/compute baselines

- fixed-grid 3D tokens (同 backbone, 同 B)
- 2D / 2.5D slice uniform sampling
- ROI-crop / coarse-to-fine (含 **detector/segmenter cost** 入账)
- no-refine (只 coarse)

Protocol baselines (消融 PCG 核心)

- no-citation / no-constraint (去掉 PCG 强制引用/约束)
- citation=top-attention tokens (decoder→encoder cross-attn 聚合 top-(K_{\max}) 作为伪引用)
- citation-only (有引用但无 verifier/无 refusal；检验“只贴引用”是否有效)

3D RRG 强 baseline

- CT2Rep 等公开 3D CT report generation 强基线。([arXiv](#))
-

7.3 预算与公平性协议 (写死)

- **B 定义**: 联合预算 ($B=B_{\text{enc}}+B_{\text{gen}}$)，并换算到 FLOPs:
 $(\text{FLOPs}_{\text{total}})=\text{FLOPs}_{\text{enc}}+\text{FLOPs}_{\text{dec}}+\text{FLOPs}_{\text{verify}}$
 - **FLOPs-matched**: 至少一组严格匹配总 FLOPs 的对比 (主结论必须在 matched setting 站得住)
 - **ROI-crop 成本**: 任何 detector/segmenter/selector 的 FLOPs 与 latency 必须计入总账；若 baseline 使用外部模型，必须报告其成本，并在 matched setting 中相应减少主模型预算
 - **latency**: 固定硬件/批大小；冷热 cache 分开；P95 基于 ≥ 1000 样本；报告 mean+P95
-

7.4 反事实三件套 (Fig3 主武器)

- **Ω -permutation**: 随机置换 (Ω_i) (cell_id) , 保持 token embedding (t_i) 不变，使 embedding 分布完全保持但空间对应关系破坏 → grounding/correctness 应显著下降
- **token-permutation**: 固定 (Ω_i) 不变，置换 (t_i) (或 token 索引) , 检验模型是否依赖正确的 token- Ω 对应
- **citation-swap**: 在同一报告内随机交换 (C_k) (保持 ($|C_k|$) 分布不变) , unsupported/overclaim 应激增
- **evidence-drop**: 删除被引用 tokens 并重生成/再验证；并做“禁引用但不删信息”的对照 (区分信息缺失 vs 引用被禁)
- **mask sanity (ReXGroundingCT)** : refine 新增 tokens 的 (Ω) 与 lesion mask overlap (IoU/Dice) 上升，证明 refine 真在“追证据”。(arXiv)

统计显著性

- paired bootstrap ($\geq 10k$ resamples) 给 95% CI；多预算多指标多重比较用 Holm 校正。

Fig3 应该画什么

- 主图: counterfactual 前后 (correctness / grounding / unsupported / overclaim) 的变化柱状或折线
- 附图: mask sanity overlap 分布 (refine 新增 tokens vs baseline)

Reviewer-check

- 缺反事实，你的 citations 会被说成装饰 ✓
 - permutation 必须“保持 embedding 分布”否则不干净 ✓
-

8. 训练与实现 (只保留最小闭环，降低风险)

8.1 训练阶段 ($M_0 \rightarrow M_3$)

- **M_0 : 协议跑通 (先能审计、能验证)**
 - 固定 coarse tokens (depth (d_0))
 - 从 reference 报告抽取 finding frames (规则+词表+模板, 附录锁死)
 - loss: slot-wise CE (frames) + 文本 NLL (由 frames 模板重写得到 target) + 引用弱监督 (鼓励选择 evidence head 高支持 token)
- **M_1 : 加入 BET refine (deterministic allocator)**
 - 引入 Algorithm 1 闭环, 产生 Fig2 多预算 Pareto
 - 不引入 RL 风险, 不改变主训练假设
- **M_2 : 接入 ReXGroundingCT 做 grounding 与反事实(arXiv)**
 - grounding consistency loss: 鼓励引用 token 的 (Ω) 与 segmentation overlap
 - 完成 Fig3 三件套
- **M_3 (可选) : learned allocator (bandit) 消融**
 - reward = issue reduction + grounding gain (归一化), 只做增益报告, 不影响主结论

8.2 可复现 artifact (强建议)

每个样本输出:

- tokens (含 Ω /cell_id) 、 citations、 verifier issues (含 rule-id、 trace、 severity)
- refusal 标记与 (q_k)
- 运行配置 (预算 B、硬件、seed) 、 数据版本 (revision) 、 split manifest、代码 commit
- refine trace: 每步 (c^t)、 ($|Delta(c^t)|$)、 issues 变化曲线 (可审计)
- ≥ 3 seeds, 附 reproducibility statement

Reviewer-check

- 不是 “我们开源” , 而是 “我们输出可审计 trace” 

9. 预期风险与应对 (提前写 rebuttal)

1. unsupported ↓ 但 miss-rate ↑ (封嘴风险)

- 把 critical miss-rate 与 refusal calibration 写成主指标; ($\tau_{\text{text}}[\text{refuse}]$) 受 miss-rate $\leq \delta$ 约束; 若 miss-rate 上升, 限制 refusal 覆盖范围并提升 evidence recall (提高 coarse coverage / 调整 stop principle)

2. verifier 太弱/太强

- 报告 verifier 强度曲线（弱→强规则集），展示协议可插拔而非绑死；taxonomy 不变，仅规则强度变化

3. latency 不稳定

- 写死测量协议（P95、冷热 cache、固定 batch），并提供 FLOPs-matched；使用增量缓存降低 refine 成本

4. 跨源泛化失败 (CT-3DRRG)

- 作为必做实验；若失败，分析 domain shift 下 allocation 崩溃点（refine 追错区域、证据图失真）并给出修复（domain-robust evidence head / conservative stop / 多源校准）
-

10. 结论（必须回到两条贡献与 B 主轴）

我们贡献：

- **BET**: 在严格预算 **B** 下的证据 tokenization (tokens 自带显式 $\Omega/\text{cell_id}$)，并给出可解释 scaling law + allocation model；
 - **PCG**: proof-carrying 生成协议，把 grounded citation + verifier + calibrated refusal 变成可审计输出与硬指标；
并在公开 3D CT 报告生成与 pixel-level grounding 上，用 Pareto 主导 + counterfactual 击穿证明其非平凡性。
-

你接下来必须“死磕”的 6 个验收条件（不满足就别谈 oral）

1. **Fig2**: 多预算 Pareto dominate (不是某个点赢)
2. **Fig2**: scaling law + allocation model 能预测最优分配 (报告 regret, 不是画曲线) ([OpenReview](#))
3. **Fig3**: Permutation / citation-swap / evidence-drop 统计显著击穿 (bootstrap+Holm)
4. **ReXGroundingCT**: citation-grounding (IoU/Dice/hit-rate) 显著提升([arXiv](#))
5. **Table1**: unsupported ↓ 的同时 critical miss-rate 不升、refusal 校准可控 (否则=封嘴)
6. **Baselines**: CT2Rep 等强基线 + FLOPs/latency matched 全齐([arXiv](#))