CONTEXTUALIZE TO GENERATE: HIERARCHICAL SEMANTIC ORGANIZATION FOR VISION–LANGUAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Image captioning translates visual content into natural language and has advanced considerably with visual representation learning and language modeling. However, different generation tasks may rely on different visual evidence. While conventional captions often focus on salient objects and their relationships, specialized reports can require detailed observations distributed across multiple regions. Simply aggregating these observations into a global representation may therefore weaken the local evidence required for generation. To better organize such evidence, here we study how local visual observations and shared context can be used together while preserving the original visual sequence, and propose **Hierarchical Semantic Contextualization (HSC)**. Specifically, HSC first uses learnable semantic anchors to extract semantic information relevant to the task from visual features under paired text guidance. Then, HSC integrates the extracted context with the original visual sequence and leverages Hierarchical Semantic Memory to progressively organize local, structural, and semantic information. In the full model, a variational residual with KL regularization further refines the contextualized representation while preserving its direct path to the generator. As a result, local observations remain available together with information collected across the input. We carefully evaluate the proposed model on whole-slide image (WSI) report generation and MS COCO captioning, covering both specialized report generation and general image captioning. The results show that the proposed model consistently improves the matched baselines,with the full configuration providing additional gains overall. Component and placement analyses further examine the contributions of the contextualization design and residual refinement. These results support the effectiveness of the proposed framework across different generation settings.

1 INTRODUCTION

Image captioning translates visual content into natural language and has advanced considerably with visual representation learning and language modeling (Vinyals et al., 2015; Anderson et al., 2018; Li et al., 2023). However, describing an image does not always require the same visual evidence. For conventional image captioning, a concise description may mainly depend on the most salient objects and their relationships. For more specialized generation tasks, the required evidence can be distributed across multiple local regions and cannot always be represented by an overall summary of the image. This is particularly evident in whole slide image (WSI) report generation, where observations from different tissue regions may jointly contribute to the final description. Simply aggregating these observations into a global representation can therefore weaken the local information required for generation.

Existing approaches organize visual evidence through different forms of aggregation. Xu et al. (2015) use attention to connect generated words with image locations, while Anderson et al. (2018) combine region features with attention for captioning. Memory mechanisms provide learned context for captioning and report generation (Cornia et al., 2020; Chen et al., 2020). HistGen combines contextual memory with hierarchical visual encoding (Guo et al., 2024). Learned query interfaces, such as Flamingo and BLIP-2, aggregate visual features into latent representations for language genera-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tion (Alayrac et al., 2022; Li et al., 2023). These approaches demonstrate that information collected beyond an individual visual location can benefit generation. However, these mechanisms typically use attention, memory, or learned latent queries primarily to summarize or retrieve contextual information for decoding. They do not explicitly study how such context can be reintegrated into a retained sequence of local visual observations so that cross-region information is introduced without replacing the evidence from which it was constructed. This distinction motivates us to consider not only what visual evidence is preserved, but also how evidence distributed across different locations is organized for generation. Retaining the original visual features keeps fine-grained observations directly accessible to the generator. However, these features alone do not explicitly capture how observations from different locations should interact when forming a coherent description. Conversely, replacing them with a globally aggregated representation can weaken access to the local evidence on which detailed descriptions depend. We therefore ask:

How can distributed visual evidence be contextualized and organized while preserving the local observations required for generation?

Based on this idea, we propose **Hierarchical Semantic Contextualization (HSC)**, which preserves the original visual sequence while progressively organizing information across visual observations. Semantic Anchoring first identifies task-relevant information from the visual features under paired-text supervision during training and uses it to contextualize the retained visual tokens. Hierarchical Semantic Memory then performs successive local, structural, and semantic aggregation, allowing information gathered at one level to inform subsequent processing. The resulting representation therefore retains fine-grained visual evidence while incorporating information collected across the input. HSC operates before an existing generation backbone and requires only the visual input at inference. In the full model, a variational residual further refines the contextualized representation. Because contextualization may also propagate information that is not consistently useful for generation, we additionally study a variational residual that regularizes only a refinement branch while preserving the direct HSC pathway to the generator. Its latent branch is regularized with a variational information bottleneck, while a direct residual path preserves the HSC representation supplied to the generator (Alemi et al., 2016).

To assess the proposed HSC, we first conduct controlled experiments on WSI report generation, where descriptions require evidence from multiple spatially distributed tissue regions. HSC consistently improves the reported generation metrics under the same generation backbone and training protocol. Further, we apply the same design to MS COCO captioning to examine whether the proposed contextualization is limited to pathology data. The improvements observed on both tasks show that organizing shared context while retaining local visual evidence can benefit substantially different image-to-text settings. To sum up, our contributions are three-fold.

(i) We formulate visual evidence organization as a balance between preserving fine-grained local observations and introducing context shared across spatially distributed regions, highlighting a limitation of representations that rely primarily on global aggregation.

(ii) We propose Hierarchical Semantic Contextualization (HSC), which constructs task-relevant context through semantic anchoring and hierarchical memory reads, and reintegrates this context into the retained visual sequence rather than replacing local observations with a global summary.

(iii) We conduct controlled experiments on WSI report generation and MS COCO captioning, and provide component analyses to examine the effectiveness and behavior of the proposed representation.

2 Related Work

Context modeling in visual-to-text generation. Visual-to-text generation requires local evidence to be interpreted together with context relevant to the target description. Early attention-based captioning models associated generated words with spatial regions (Xu et al., 2015; Anderson et al., 2018), while memory-based generators introduced persistent visual or semantic context into the decoding process (Cornia et al., 2020; Chen et al., 2020). Similar ideas have been developed for medical report generation, where structured visual evidence and cross-modal context are particularly important (Jing et al., 2018; Guo et al., 2024). These methods demonstrate the value of contextual information for generation, but mainly incorporate context through the existing encoder–decoder

interaction. Our work instead focuses on the visual representation before the downstream generation backbone: HSC constructs shared, description-relevant context and reinjects it into the retained visual sequence. The HSC memories introduced here are separate from the downstream HistGen modules, which remain unchanged in the controlled comparisons.

Learned-query interfaces and visual organization. A related line of work reorganizes dense visual inputs through learned latent representations. Perceiver maps large inputs into a latent array through cross-attention (Jaegle et al., 2021), and Slot Attention iteratively forms object-centric slots (Locatello et al., 2020). Flamingo's Perceiver Resampler (Alayrac et al., 2022) and BLIP-2's Q-Former (Li et al., 2023) similarly use learned queries to mediate vision–language interaction. Other methods explicitly reduce the visual sequence through token learning, pruning, or merging (Ryoo et al., 2021; Rao et al., 2021; Liang et al., 2022; Bolya et al., 2022), and recent work has studied task-related compression in multimodal models (Lei et al., 2026). HSC differs in that the query responses are not used to replace the visual sequence. Instead, their pooled response is returned to the original N visual positions, after which HSM constructs additional memory-conditioned context. Thus, the anchor bank controls context extraction rather than the number of visual tokens passed forward.

Information bottleneck and variational regularization. The information bottleneck principle characterizes representations that retain information relevant to a target while limiting unnecessary information about the input (Tishby et al., 2000). Deep VIB makes this principle tractable using stochastic latent variables and variational bounds (Alemi et al., 2016). Related formulations have been studied in multi-view learning (Federici et al., 2020) and multimodal settings, including unpaired image-to-speech synthesis (Ma et al., 2019) and image–text attribution (Wang et al., 2023). Our use of VIB is more restricted than a replacement bottleneck: the contextualized representation retains a deterministic path to the generator, while only an auxiliary residual latent is regularized. We use this formulation to study how an information-constrained residual interacts with semantic organization, and compare its placement before and after hierarchical contextualization. Appendix A.2 gives the corresponding variational interpretation and explicitly states the scope of the bound.

3 Method

3.1 Problem Setting and Structure-First Formulation

Given paired visual input and target text (X, Y), where $Y = (y_1, \ldots, y_T)$, a visual encoder extracts $V = f_v(X) \in \mathbb{R}^{N \times d}$. A language decoder models

$$p_\theta(Y \mid R) = \prod_{t=1}^{T} p_\theta(y_t \mid y_{<t}, R), \tag{1}$$

where R denotes the visual representation supplied to the generation backbone.

The visual input generally contains more information than is expressed in the target text. We therefore separate two roles: *semantic organization*, which constructs task-relevant context from the visual sequence, and *information-constrained refinement*, which regularizes an auxiliary residual update after contextualization. Let

$$S = F_\psi(V) \tag{2}$$

denote the contextualized representation produced by HSC, and let $Z \sim q_\phi(Z \mid S)$ denote a stochastic residual latent. The final representation is

$$R = S + \alpha g(Z), \qquad \alpha \ll 1. \tag{3}$$

From an information-bottleneck perspective, we organize the learning objective as

$$\max_{\psi,\phi,\theta} \; I(R;Y) - \beta I(Z;S). \tag{4}$$

The first term favors information useful for predicting the target text, whereas the second controls the capacity of the residual latent. Importantly, the direct path from S to R remains intact, so the information constraint applies to the residual branch rather than the complete representation. Standard

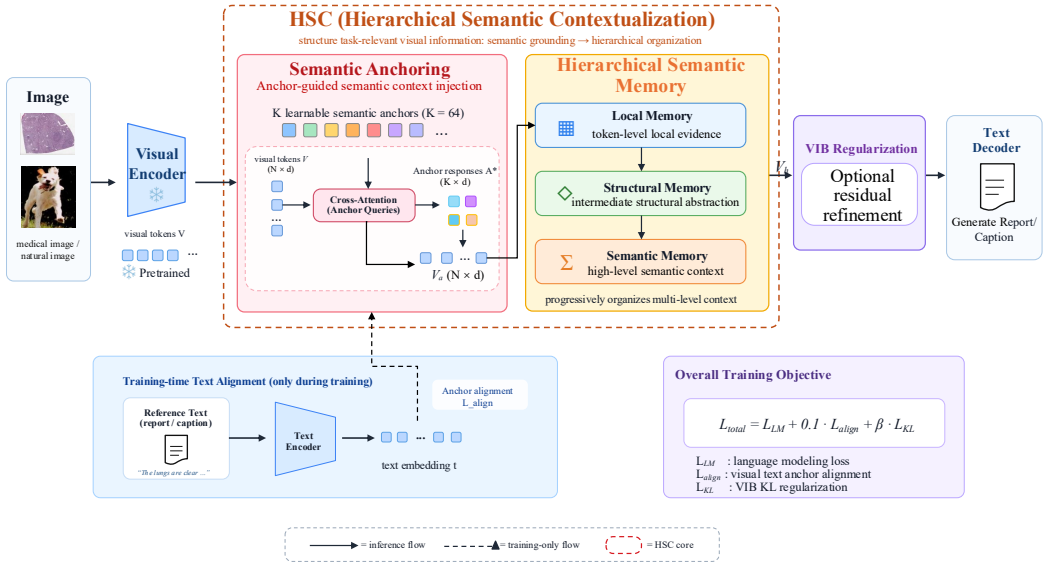


Figure 1: **Learning framework of HSC.** HSC contextualizes the visual representation in two successive steps. In Step 1, semantic anchoring extracts task relevant context from the visual tokens and injects it back into the original visual sequence. In Step 2, hierarchical semantic memory progressively organizes the enriched representation through local, structural, and semantic context. An optional VIB branch provides residual regularization before the contextualized visual representation is passed to the text decoder. Text alignment is used only during training.

variational bounds connect these terms to autoregressive prediction and Gaussian KL regularization; the corresponding derivation is given in Appendix A.2.

Our structure-first instantiation follows

$$V \xrightarrow{\text{HSC}} S \xrightarrow{\text{residual VIB}} R, \tag{5}$$

where HSC establishes task-relevant context before the variational residual is applied. This objective motivates the representation design, while the effectiveness of the architectural instantiation and the residual placement are evaluated empirically rather than assumed theoretically. Figure 1 summarizes the full computation.

3.2 Hierarchical Semantic Contextualization

Semantic Anchoring. A learnable anchor bank $A \in \mathbb{R}^{K\times d}$ queries projected visual features through multi-head attention:

$$A^* = \mathrm{MHA}(A, W_v V, W_v V), \qquad c_A = \frac{1}{K}\sum_{k=1}^{K} A^*_k, \qquad V_A = V + \mathbf{1}_N c_A^\top. \tag{6}$$

The anchors are learnable coordinates rather than predefined semantic categories. Their responses depend on the current input, so c_A provides shared, input-dependent context. Unlike query resampling, this operation does not replace the visual sequence: $V_A \in \mathbb{R}^{N\times d}$ retains all N positions and preserves the original features through the residual path.

During training, paired target text guides the anchor space through an auxiliary alignment objective. Let $E_Y = W_t\,\mathrm{Emb}(Y)$ denote projected target-token embeddings. Normalized visual-to-anchor and text-to-anchor similarities S^v, S^t, computed with temperature τ, enter

$$\mathcal{L}_{\text{align}} = \mathcal{A}(S^v, S^t). \tag{7}$$

This supervision encourages the extracted context to reflect information relevant to the target description. Reference text is absent from the inference-time visual pathway.

Hierarchical Semantic Memory. Semantic Anchoring supplies shared context, but evidence in the visual sequence remains distributed across different abstraction levels. HSM therefore maintains local, structural, and semantic memories M_L, M_S, M_M. The anchor-enriched visual sequence first queries the local memory; the pooled local response queries structural memory, whose response then queries semantic memory:

$$\begin{aligned} H_L &= \mathrm{MHA}(W_L V_A, M_L, M_L), \quad \bar{H}_L = \frac{1}{N}\sum_i H_{L,i}, \\ H_S &= \mathrm{MHA}(W_S \bar{H}_L, M_S, M_S), \quad H_M = \mathrm{MHA}(W_M H_S, M_M, M_M). \end{aligned} \tag{8}$$

The hierarchy is defined by these query dependencies: the local read operates at each visual position, whereas the subsequent reads act on summaries formed by the preceding level. Rather than retaining only the deepest response, we combine all three levels. With $h_H = W_o[\bar{H}_L; H_S; H_M]$ and $\bar{V}_A = N^{-1}\sum_i V_{A,i}$,

$$h = \mathrm{LN}(h_H + \bar{V}_A), \qquad S = V_A + \mathbf{1}_N h^\top. \tag{9}$$

Thus, each output token can be written as $S_i = V_i + c_A + h$: the original local evidence is retained while anchor-derived and memory-derived contexts are added. HSC therefore constructs task-relevant semantic organization without reducing the visual sequence to the K anchor responses.

3.3 Information-Constrained Residual Refinement

After contextualization, the full configuration adds a variational residual branch. For each s_i, the posterior parameters and training-time sample are defined by

$$\begin{aligned} q_\phi(z_i \mid s_i) &= \mathcal{N}\big(\mu_i, \mathrm{diag}(\sigma_i^2)\big), \quad \mu_i = W_\mu s_i, \\ \log \sigma_i^2 &= \mathrm{clip}(W_\sigma s_i, -8, 8), \quad z_i = \mu_i + \sigma_i \odot \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, I). \end{aligned} \tag{10}$$

At inference, the posterior mean $z_i = \mu_i$ is used.

With a standard Gaussian prior $r(Z) = \mathcal{N}(0, I)$, the variational regularizer is

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{2Nd_z}\sum_{i=1}^{N}\sum_{j=1}^{d_z}\left(\mu_{ij}^2 + \sigma_{ij}^2 - 1 - \log \sigma_{ij}^2\right). \tag{11}$$

The standard variational decomposition gives

$$\mathbb{E}_{p(S)} D_{\mathrm{KL}}(q_\phi(Z \mid S) \| r(Z)) = I(Z; S) + D_{\mathrm{KL}}(q_\phi(Z) \| r(Z)) \geq I(Z; S), \tag{12}$$

so minimizing the KL term constrains an upper bound on the information capacity of the residual latent.

The latent does not replace S. Instead, it is added through $R = S + \alpha g(Z)$ with $\alpha = 0.01$. The output map g is zero-initialized, so the branch begins as an identity update. The direct path guarantees that contextualized features remain available to the generator even though the residual latent is regularized. Accordingly, the KL term should be interpreted as a constraint on the residual refinement, not as evidence that all information in R is compressed.

The practical objective and its autoregressive language-modeling term are

$$\begin{aligned} \mathcal{L}_{\mathrm{total}} &= \mathcal{L}_{\mathrm{LM}} + 0.1\mathcal{L}_{\mathrm{align}} + \beta\mathcal{L}_{\mathrm{KL}}, \\ \mathcal{L}_{\mathrm{LM}} &= -\mathbb{E}\sum_{t=1}^{T} \log p_\theta(y_t \mid y_{<t}, R). \end{aligned} \tag{13}$$

The HSC-only configuration omits the residual branch and its KL term. Keeping the branch while setting $\beta = 0$ is a distinct control used to separate the residual architecture from its information constraint.

3.4 Structure-First Residual Placement

The default full model regularizes the HSC output S. To test whether the available semantic organization affects the usefulness of the constrained update, we compare two placements using the same variational parameterization, residual scale, and KL normalization. The early variant applies the residual immediately after Semantic Anchoring:

$$V_A \to V_A + \alpha g(Z_A) \to \mathrm{HSM}, \tag{14}$$

whereas the structure-first variant follows

$$V_A \rightarrow \text{HSM} \rightarrow S + \alpha g(Z_S). \tag{15}$$

The comparison therefore changes the representation receiving the information-constrained update, rather than the form of the residual regularizer itself. We use this experiment to test the ordering hypothesis; the formulation does not assume that later insertion is universally superior.

HSC is inserted after visual feature extraction and before the existing generation backbone. For WSI, HistGen's visual-memory interaction and hierarchical visual encoder remain downstream before report decoding. For COCO, the same contextualization-and-refinement interface precedes the captioning generator. Backbone modules are unchanged within each controlled comparison.

4 Experiments

Our experiments evaluate the two roles of the proposed structure-first framework: semantic organization through HSC and information-constrained residual refinement. We use WSI report generation as the primary setting, where we first compare with existing methods and then isolate each contribution under a matched generation pipeline. We further examine HSC components and residual placement before applying the same representation design to MS COCO captioning. We then summarize computational efficiency and hyperparameter behavior, while the appendix reports complete metric tables, diagnostic controls, and data-efficiency results.

4.1 Implementation Details

Datasets and evaluation. For WSI report generation, we use the benchmark distributed with HistGen (Guo et al., 2024), containing 7,690 cases under the benchmark partition, of which 6,152 are used for training. Each slide is represented by precomputed patch features, and reports are truncated to at most 100 tokens. For general image captioning, we use MS COCO (Lin et al., 2014) with the Karpathy split (Karpathy & Fei-Fei, 2015). The WSI and COCO models are trained independently. We report BLEU-1–BLEU-4 (Papineni et al., 2002), METEOR (Banerjee & Lavie, 2005), and ROUGE-L (Lin, 2004). CIDEr (Vedantam et al., 2015) and SPICE (Anderson et al., 2016) are additionally reported for COCO. Results are averaged over three random seeds. Checkpoint and hyperparameter selection follow a fixed BLEU-4-based selection protocol. All reported comparisons use the evaluation partition specified by their corresponding experiments.

Configurations. HSC comprises Semantic Anchoring and Hierarchical Semantic Memory (HSM). The complete configuration, denoted as **Ours**, further applies the variational residual after HSC. The default model uses 64 semantic anchors and a latent width of 512. For WSI, the proposed representation is inserted before the existing HistGen generation backbone, which is kept unchanged in controlled comparisons. WSI and COCO models are trained for 40 and 50 epochs, respectively. Full optimization and implementation settings are provided in Appendix A.4.

4.2 Performance Comparison

Comparison with existing methods. We first ask whether the proposed representation improves WSI report generation relative to existing approaches. Table 1 compares Ours with representative captioning and report generation methods. Ours obtains the highest value in all six reported columns. In particular, BLEU-4 increases from 0.158 for HistGen to 0.178, while METEOR increases from 0.163 to 0.176. The improvement is therefore not confined to a single generation metric.

Controlled comparison. The comparison above does not by itself isolate the contribution of the proposed representation. We therefore compare HistGen, HSC, and Ours while keeping the visual features, downstream backbone, optimization budget, checkpoint selection, and decoding procedure fixed. Table 2 shows that HSC raises BLEU-4 from 0.158 to 0.175 and ROUGE-L from 0.314 to 0.338. The complete model gives a further increase in the underlying means, reaching 0.408 BLEU-1 and 0.178 BLEU-4. Across all six metrics, Ours achieves the highest underlying mean, with some differences obscured by rounding to three decimal places. Specifically, relative to the matched HistGen baseline, the complete model improves BLEU-4 by 0.020 absolute, of which HSC accounts

Table 1: **Comparison with existing methods on WSI.** B1–B4, M, and R-L denote BLEU-1–BLEU-4, METEOR, and ROUGE-L. Our results are averaged over three random seeds.

Method	B1	B2	B3	B4	M	R-L
Show&Tell	0.189	0.094	0.056	0.039	0.070	0.165
UpDownAttn	0.320	0.206	0.147	0.112	0.131	0.271
Transformer	0.365	0.248	0.185	0.139	0.145	0.312
M2Transformer	0.321	0.213	0.152	0.112	0.131	0.266
R2Gen	0.274	0.166	0.128	0.107	0.121	0.248
HistGen	0.375	0.261	0.199	0.158	0.163	0.314
Ours	**0.408**	**0.289**	**0.222**	**0.178**	**0.176**	**0.338**

Table 2: **Controlled comparison on WSI.** All configurations use the same visual features, downstream backbone, training protocol, checkpoint selection, and decoding procedure.

Method	B1	B2	B3	B4	M	R-L
HistGen	0.375	0.261	0.199	0.158	0.163	0.314
HistGen+HSC	0.406	0.288	0.222	0.175	0.174	0.338
Ours	**0.408**	**0.289**	**0.222**	**0.178**	**0.176**	**0.338**

for 0.017. The dominant improvement is therefore introduced by HSC, while the variational residual provides a smaller increase in the mean. We therefore interpret the residual as a complementary refinement within this pipeline rather than as a large standalone improvement.

4.3 Empirical Analysis

HSC components. We first ask which parts of HSC account for its improvement by removing Semantic Anchoring and HSM separately. Table 3 shows that removing Semantic Anchoring reduces BLEU-4 from 0.175 to 0.161, while removing HSM gives a larger reduction to 0.135. An independently implemented HSM-only control reaches 0.173, indicating that hierarchical aggregation accounts for a substantial part of the improvement. Since this control uses different routing from the removal variants, we treat it as a separate diagnostic rather than as the exact complement of the no-anchor configuration.

Contextualization controls. We next test whether the gain can be explained by additional contextualization alone. A parameter-matched flat control retains the same memory and attention capacity as HSC but removes the progressive cross-level query dependencies. It reaches 0.159 BLEU-4, substantially below HSC at 0.175. Moreover, removing the training-time text-alignment objective reduces BLEU-4 to 0.160. These results show that hierarchical organization provides the main architectural contribution, while text-guided Semantic Anchoring supplies complementary task-relevant context.

Residual placement. The variational residual is intended to refine an already contextualized representation. To test whether this ordering matters, we apply the same residual either immediately after Semantic Anchoring or after HSM. Early insertion gives 0.172 BLEU-4, compared with 0.178 when the residual is applied after HSM. Early insertion also falls below HSC without VIB. Within this pipeline, the result therefore favors applying the constrained residual after semantic contextualization.

Anchor response visualization. To provide an intuitive understanding of Semantic Anchoring, here we visualize the responses of selected semantic anchors on three WSI cases. Specifically, since the stored features do not contain slide coordinates, we project the mapped features before Semantic Anchoring into two dimensions using PCA and use the same coordinates in all anchor panels. We evenly select three cases from reports containing 35–65 whitespace tokens. For each case, we first choose the anchor with the highest coefficient of variation and then greedily select three additional

Table 3: **Empirical analysis on WSI.** The upper block examines HSC components and matched contextualization controls. Flat contextualization retains the same memory and attention capacity as HSC but removes progressive cross-level dependencies. The lower block changes only the placement of the variational residual.

Configuration	B1	B4
HSC components		
HSC w/o Semantic Anchoring	0.401	0.161
HSC w/o HSM	0.341	0.135
HSM only	0.403	0.173
Flat contextualization	0.389	0.159
HSC, no alignment	0.397	0.160
HSC	**0.406**	**0.175**
Residual placement		
VIB after Semantic Anchoring	0.400	0.172
Ours: VIB after HSM	**0.408**	**0.178**

Table 4: **Evaluation on the MS COCO Karpathy split.** All configurations use the same captioning backbone, data split, training schedule, and decoding procedure.

Method	B1	B4	M	R-L	CIDEr	SPICE
HistGen	0.672	0.266	0.229	0.496	0.812	0.155
HistGen + HSC	0.674	0.271	0.229	0.498	0.821	0.156
Ours	**0.677**	**0.274**	**0.231**	**0.501**	**0.827**	**0.159**

anchors with high variation and low correlation. Meanwhile, we show excerpts from the reference report and the reports generated by HistGen and Ours for comparison. The results are shown in Figure 2. We can see that the response patterns vary across anchors and cases, which provides an intuitive view of how the learned anchors distribute attention among mapped patches. Note that these panels illustrate responses in feature space rather than spatial localization or pathology annotations.

4.4 Evaluation on General Image Captioning

The WSI setting contains highly distributed visual evidence. We therefore test whether the same representation design is also useful in a standard image captioning setting. On MS COCO, the captioning backbone, data split, training schedule, and decoding procedure are held fixed across the three configurations. Table 4 shows an overall improvement under the complete model. BLEU-4 increases from 0.266 for the baseline to 0.271 with HSC and to 0.274 with Ours. ROUGE-L increases from 0.496 to 0.501, while METEOR reaches 0.231. The effect is smaller than on WSI, but it is also observed in this non-pathology captioning setting. Because the models are trained independently, this experiment tests applicability across generation settings rather than zero-shot transfer. The complete results in Table B1 show that Ours attains the highest mean across all eight metrics. HSC improves seven metrics and matches the baseline METEOR value of 0.229. Specifically, CIDEr increases from 0.812 to 0.827, while SPICE increases from 0.155 to 0.159. These consistent changes show that the gain is not driven by a single metric.

4.5 Efficiency and Sensitivity

Computational efficiency. Table B4 shows that HSC raises WSI BLEU-4 from 0.158 to 0.175, adding 8.68M parameters and 0.04 seconds per report. Ours adds 0.10M parameters and 0.02 seconds over HSC to reach 0.178. HSC therefore accounts for most of the gain and cost, while the residual adds a smaller BLEU-4 increase.

Hyperparameter behavior. Tables B6 and B7 show that both sweeps peak at 0.178 BLEU-4 under the selected settings $K = 64$ and $\beta = 3 \times 10^{-3}$. Setting $K = 128$ lowers the mean to 0.165, and no other tested KL coefficient improves on these settings; both grids favor moderate values.

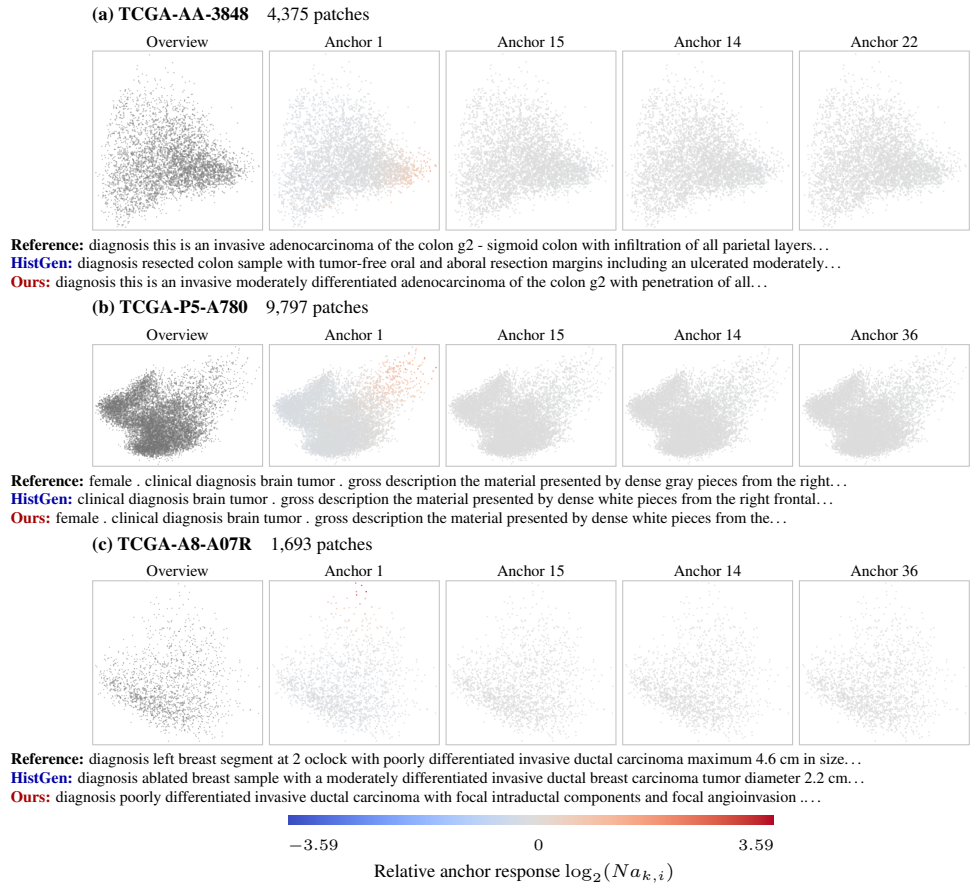


Figure 2: Semantic anchor responses and report excerpts for three WSI cases. Rows project all mapped features before Semantic Anchoring, using all points but no slide coordinates. The first anchor has the highest coefficient of variation; three others are chosen greedily for high variation and low correlation. Cases evenly sample reports containing 35–65 whitespace tokens. Colors show the full symmetric range of $\log_2(Na_{k,i})$ without clipping, with N the patch count and $a_{k,i}$ the normalized attention averaged over heads; uniform attention is zero. The panels show feature space, not slide locations or pathology annotations.

5 Conclusion and Limitations

In this paper, we study how to better organize visual evidence for vision–language generation while preserving the local information required for detailed prediction. To tackle this problem, we propose a hierarchical semantic contextualization framework which enriches the original visual representation with task-relevant context and progressively organizes information before generation. Extensive experiments on specialized report generation and general image captioning tasks demonstrate the effectiveness of the proposed method across different generation settings. However, our study is limited in several aspects. First, the current experiments are conducted on a limited number of datasets and generation backbones, and broader evaluations are needed to further examine its generality. Second, the current evaluation mainly relies on standard generation metrics, which may not fully reflect factual correctness and semantic faithfulness. In the future, we hope to extend the framework to broader architectures and tasks and investigate more reliable evaluation protocols for vision–language generation.

Reproducibility Statement

Section 3 and Appendix A describe the architecture and variational objective. Section 4.1 specifies the training, checkpoint-selection, and decoding protocol. Repeated default-configuration results are linked to a common value registry. The appendix provides additional settings and analyses.

Ethics Statement

The medical generation experiments assess model outputs on research benchmarks. Generation metrics and attention maps do not establish diagnostic safety, and generated reports are not intended for autonomous clinical use. Clinical deployment would require domain-appropriate assessment and oversight.

AI Usage Statement

Generative AI tools were used only for language editing and limited assistance with manuscript organization. All scientific content, experiments, analyses, and conclusions were independently developed and verified by the authors.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, pp. 382–398. Springer, 2016.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 6077–6086. IEEE, 2018.

Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72, 2005.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022.

Zhihong Chen, Yan Song, Tsung-Hui Chang, and Xiang Wan. Generating radiology reports via memory-driven transformer. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 1439–1449, 2020.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10575–10584. IEEE, 2020.

Marco Federici, Anjan Dutta, Patrick Forré, Nate Kushman, and Zeynep Akata. Learning robust representations via multi-view information bottleneck. *arXiv preprint arXiv:2002.07017*, 2020.

Zhengrui Guo, Jiabo Ma, Yingxue Xu, Yihui Wang, Liansheng Wang, and Hao Chen. Histgen: Histopathology report generation via local-global feature encoding and cross-modal context interaction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 189–199. Springer, 2024.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pp. 4651–4664. PMLR, 2021.

Baoyu Jing, Pengtao Xie, and Eric Xing. On the automatic generation of medical imaging reports. In *Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 2577–2586, 2018.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128–3137, 2015.

Lei Lei, Jie Gu, Xiaokang Ma, Chu Tang, Jingmin Chen, and Tong Xu. Task-related token compression in multimodal large language models from an explainability perspective. In *International Conference on Learning Representations*, volume 2026, pp. 112724–112746, 2026.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PmLR, 2023.

Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. *arXiv preprint arXiv:2202.07800*, 2022.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Shuang Ma, Daniel McDuff, and Yale Song. Unpaired image-to-speech synthesis with multimodal information bottleneck. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7597–7606. IEEE, 2019.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318. Association for Computational Linguistics, 2002. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040/.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021.

Michael Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Tokenlearner: Adaptive space-time tokenization for videos. *Advances in neural information processing systems*, 34:12786–12797, 2021.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4566–4575, 2015.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164, 2015.

Ying Wang, Tim GJ Rudner, and Andrew G Wilson. Visual explanations of image-text representations via multi-modal information bottleneck attribution. *Advances in Neural Information Processing Systems*, 36:16009–16027, 2023.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057. PMLR, 2015.

A Extended Method Details

A.1 Expanded Hierarchical Semantic Contextualization

The anchor bank $A \in \mathbb{R}^{K\times d}$ produces responses $A^* \in \mathbb{R}^{K\times d}$ through the attention operation in Eq. (6). Their mean provides a shared, input-dependent context that is returned to the original visual sequence. The number of anchors therefore controls the context-extraction interface rather than the number of visual tokens passed to the downstream generator.

HSM maintains local, structural, and semantic memories M_L, M_S, and M_M. The three stages are defined by their query dependencies. The local memory is queried at each visual position. Its pooled response then queries structural memory, and the structural response queries semantic memory. The three responses are combined and projected back to the visual width before being returned to the retained visual sequence. Accordingly, HSC preserves the original visual positions while enriching them with shared anchor context and hierarchical memory context.

Training-time alignment. Projected target-token embeddings and visual features form normalized similarities with the anchor bank. The auxiliary agreement term $\mathcal{A}(S^v, S^t)$ is applied with weight $\lambda_{\text{align}} = 0.1$. This term supplies task-related supervision to the anchor space during training. The reference text is not part of the inference-time visual pathway.

Flat contextualization control. To isolate the contribution of progressive memory dependencies, we construct a parameter-matched flat contextualization control. The control retains the same Semantic Anchoring module, three memory banks, attention blocks, hidden widths, output projection, residual reinjection, alignment objective, and training protocol as HSC. The only change is the dependency among memory reads. Specifically, letting $\bar{V}_A = N^{-1}\sum_i V_{A,i}$, the local read remains

$$H_L^{\text{flat}} = \text{MHA}(W_L V_A, M_L, M_L), \qquad \bar{H}_L^{\text{flat}} = \frac{1}{N}\sum_i H_{L,i}^{\text{flat}},$$

while the structural and semantic memories independently query the same anchor-enriched visual summary,

$$H_S^{\text{flat}} = \text{MHA}(W_S \bar{V}_A, M_S, M_S), \qquad H_M^{\text{flat}} = \text{MHA}(W_M \bar{V}_A, M_M, M_M).$$

The three responses are combined using the same output projection as HSC. Thus, the flat control has the same parameter count and number of attention operations as HSC, but removes the progressive $\bar{H}_L \to H_S \to H_M$ dependency.

A.2 Information-Theoretic Interpretation of the Residual Regularizer

The language-modeling and Gaussian KL terms admit an information-bottleneck interpretation. A corresponding conceptual objective is

$$\max_{\psi,\phi,\theta} I(R;Y) - \beta I(Z;S), \tag{16}$$

where predictive information is associated with the final representation R, while the constrained latent is Z. Because R also receives S through a direct deterministic path, this interpretation concerns the residual branch rather than the complete representation.

Predictive lower bound. Under the joint distribution induced by paired inputs and the stochastic encoder,

$$
\begin{aligned}
I(R;Y) &= H(Y) + \mathbb{E}_{p(R,Y)} \log p(Y \mid R) \\
&= H(Y) + \mathbb{E}_{p(R,Y)} \log p_\theta(Y \mid R) \\
&\quad + \mathbb{E}_{p(R)} D_{\mathrm{KL}}\left(p(Y \mid R) \| p_\theta(Y \mid R)\right) \\
&\geq H(Y) + \mathbb{E}_{p(R,Y)} \log p_\theta(Y \mid R).
\end{aligned}
\tag{17}
$$

Since $H(Y)$ does not depend on the model, maximizing the lower bound yields the autoregressive language-modeling objective used during training.

Residual-capacity upper bound. Let

$$
q_\phi(Z) = \int p(S) q_\phi(Z \mid S)\, dS
$$

denote the aggregate posterior and let $r(Z)$ be the standard Gaussian prior. Then

$$
\begin{aligned}
&\mathbb{E}_{p(S)} D_{\mathrm{KL}}\left(q_\phi(Z \mid S) \| r(Z)\right) \\
&= I(Z;S) + D_{\mathrm{KL}}\left(q_\phi(Z) \| r(Z)\right) \geq I(Z;S).
\end{aligned}
\tag{18}
$$

The KL term used in the implementation is normalized by the number of visual positions and latent dimensions. The coefficient β therefore applies to a per-position, per-dimension regularizer rather than the unnormalized sum.

Scope of the bound. The residual update

$$
R = S + \alpha g(Z)
$$

preserves a direct path from S to R. Bounding $I(Z;S)$ therefore does not establish a bound on $I(R;S)$ and does not imply that the complete representation is compressed. The residual branch should instead be interpreted as a constrained stochastic refinement of the HSC output.

A.3 Early and Late Residual Placement

Let F_{HSM} denote the hierarchical transformation after Semantic Anchoring. The early placement is

$$
Z_A \sim q_\phi(Z_A \mid V_A), \qquad \widetilde{V}_A = V_A + \alpha g(Z_A), \qquad R_{\mathrm{early}} = F_{\mathrm{HSM}}(\widetilde{V}_A),
\tag{19}
$$

whereas the default structure-first placement is

$$
S = F_{\mathrm{HSM}}(V_A), \qquad Z_S \sim q_\phi(Z_S \mid S), \qquad R_{\mathrm{late}} = S + \alpha g(Z_S).
\tag{20}
$$

The two variants use the same latent parameterization, residual scale, Gaussian KL normalization, downstream backbone, and training procedure. Only the representation receiving the residual update is changed.

Residual-path behavior. For any compatible norm,

$$
\|R - S\| = |\alpha| \, \|g(Z)\|.
\tag{21}
$$

The output map is zero-initialized, so the residual branch begins as an identity update. The scale $\alpha = 0.01$ controls the contribution of the branch, but does not by itself provide a fixed upper bound on the size of the learned correction.

A.4 Integration and Implementation Settings

For WSI reporting, the transformed visual sequence enters the existing HistGen visual-memory interaction and hierarchical visual encoder before autoregressive report decoding. HSC and HSM are additional representation modules and are distinct from the downstream HistGen components. For COCO, the same representation interface precedes the captioning generator.

Table A1: **Implementation settings.** The residual output map is zero-initialized and posterior-mean inference is used for the variational branch.

Setting	WSI	MS COCO
Learning rate	10^{-4}	10^{-4}
Anchor count K	64	64
Alignment temperature τ	0.07	0.07
Alignment weight λ_{align}	0.1	0.1
Latent width d_z	512	512
Residual scale α	0.01	0.01
Log-variance clipping	$[-8, 8]$	$[-8, 8]$
Memory sizes (n_L, n_S, n_M)	(256, 128, 64)	
Memory width d_h	512	

B Additional Experimental Results and Analyses

B.1 Complete MS COCO Results

Table B1 reports the complete set of COCO generation metrics corresponding to the compact comparison in the main paper.

Table B1: **Complete MS COCO results.** All configurations use the same data split, captioning backbone, training schedule, and decoding procedure.

Method	B1	B2	B3	B4	M	R-L	CIDEr	SPICE
HistGen	0.672	0.494	0.362	0.266	0.229	0.496	0.812	0.155
HistGen + HSC	0.674	0.499	0.364	0.271	0.229	0.498	0.821	0.156
Ours	**0.677**	**0.501**	**0.369**	**0.274**	**0.231**	**0.501**	**0.827**	**0.159**

Ours achieves the highest score in all eight metrics. Relative to the baseline, BLEU-4 increases from 0.266 to 0.274, while CIDEr increases from 0.812 to 0.827. HSC improves most metrics, and the full configuration reports a higher mean in every column.

B.2 Complete HSC Component Analysis

Table B2 provides the complete B1 and B4 component comparison under the same protocol as the main paper.

Table B2: **Complete HSC component analysis on WSI.**

Configuration	B1	B4
HSC	0.406	0.175
No Semantic Anchoring	0.401	0.161
No HSM	0.341	0.135
HSM only	0.403	0.173
Ours	**0.408**	**0.178**

The complete model obtains the highest score in both reported columns. Among the HSC ablations, removing HSM causes the largest drop, consistent with the main-paper analysis that hierarchical aggregation accounts for most of the contextualization gain.

B.3 Complete Residual Placement Results

Table B3 reports the four generation metrics for the two residual placements together with the HSC reference.

Table B3: **Effect of variational residual placement on WSI.** Results are reported as mean ± standard deviation over three random seeds.

Configuration	B1	B4	M	R-L
HSC, no VIB	0.406 ± 0.002	0.175 ± 0.001	0.174 ± 0.001	0.338 ± 0.002
VIB after Semantic Anchoring	0.400 ± 0.002	0.172 ± 0.001	0.173 ± 0.001	0.333 ± 0.002
VIB after HSM	$\mathbf{0.408 \pm 0.003}$	$\mathbf{0.178 \pm 0.002}$	$\mathbf{0.176 \pm 0.001}$	$\mathbf{0.338 \pm 0.003}$

Applying the residual after HSM gives the strongest overall performance across the four metrics. Early placement is lower than HSC without VIB in every column, supporting structure-first contextualization before residual refinement.

B.4 Generation Quality and Computational Overhead

We profile the three WSI configurations under the same inference scope and decoding policy. Offline feature extraction is excluded. This comparison measures the additional cost of context construction and residual refinement without assuming that either component reduces computation.

Table B4: **Parameter overhead, inference time, and generation performance on WSI.** Added parameters are reported in millions.

Method	Added params (M) ↓	Peak memory (GB) ↓	Time (s/report) ↓	B4 ↑
HistGen	0.00	0.51	0.31	0.158
HSC	8.68	0.54	0.35	0.175
Ours	8.78	0.56	0.37	**0.178**

HSC improves BLEU-4 by 0.017 over HistGen with 8.68M additional parameters and 0.04 seconds of additional inference time per report. The variational residual adds 0.10M parameters and 0.02 seconds per report over HSC while raising BLEU-4 by a further 0.003.

B.5 Diagnostic Controls

Routing-specific controls. Table B5 reports compact-only, anchor-only, and VIB-only implementations. These models use different internal routing and are therefore treated as diagnostic controls rather than direct component removals of HSC.

Table B5: **Routing-specific controls on WSI.**

Configuration	B1	B4	M	R-L
Compact only	0.355	0.143	0.152	0.294
Anchor only	0.337	0.129	0.145	0.276
VIB only	0.324	0.112	0.135	0.261

Compact-only is the strongest of these isolated controls, but its BLEU-4 of 0.143 remains below the matched HSC result of 0.175. The individual routing operations therefore do not reproduce the gain of the complete hierarchical contextualization pathway.

B.6 Hyperparameter Sensitivity

Number of semantic anchors. We vary K while keeping the remaining model and training settings fixed. Changing K changes the capacity of the context-extraction interface but does not itself reduce the number of retained visual positions.

Table B6: **Sensitivity to the number of semantic anchors on WSI.** All configurations follow the same WSI protocol.

K	B1	B4	M	R-L
16	0.373	0.152	0.146	0.304
32	0.389	0.161	0.160	0.321
64	**0.408**	**0.178**	**0.176**	**0.338**
128	0.392	0.165	0.162	0.324

KL regularization strength. We vary the coefficient β while retaining the same residual architecture. The $\beta = 0$ configuration retains the residual branch but removes its weighted KL penalty. HSC without VIB is therefore a different control.

Table B7: **Sensitivity to the KL coefficient on WSI.** All configurations follow the same WSI protocol.

β	B1	B4	M	R-L
HSC, no VIB	0.406	0.175	0.174	0.338
10^{-5}	0.402	0.171	0.169	0.332
10^{-4}	0.405	0.173	0.171	0.332
10^{-3}	0.407	0.176	0.175	0.336
$\mathbf{3 \times 10^{-3}}$	**0.408**	**0.178**	**0.176**	**0.338**
10^{-2}	0.402	0.169	0.170	0.330

Both sweeps favor the default settings: $K = 64$ gives the highest scores in the anchor-count study, and $\beta = 3 \times 10^{-3}$ gives the highest or tied-highest score in every reported metric of the KL-coefficient study.

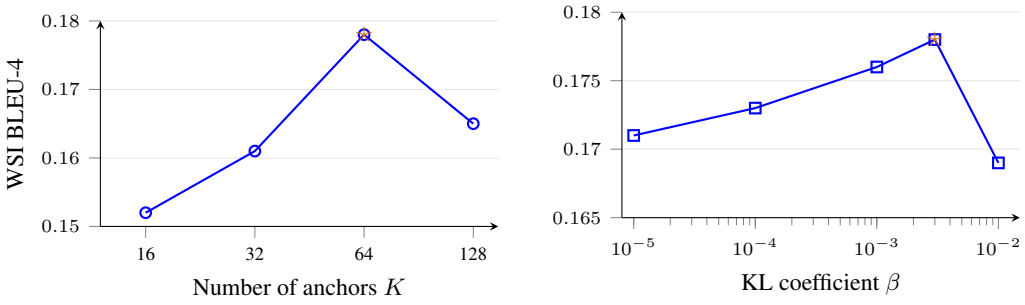


Figure B1: **Hyperparameter sensitivity on WSI.** Within each sweep, every point reports the three-seed mean BLEU-4 while all other settings are fixed. Left: number of semantic anchors. Right: positive KL coefficients. Orange stars indicate the default settings, $K = 64$ and $\beta = 3 \times 10^{-3}$.

B.7 Data-Efficiency Analysis

We examine generation performance when the amount of WSI training data is reduced. The same protocol is used across all training fractions, and this analysis is reported separately from the three-seed comparisons in the main paper.

Table B8: **We evaluate the full model under reduced training fractions.** The comparison protocol is unchanged across training fractions.

Training fraction	BLEU-4
10%	0.111
25%	0.129
50%	0.136
75%	0.148
100%	**0.178**

Performance rises steadily as additional supervision is introduced. The result is used as a data-efficiency analysis and is not combined numerically with the main three-seed comparisons.