MELBA Journal Machine Learning for Biomedical Imaging

TRACE: Tissue and Report Atlas of Computational Embeddings

Siddhesh **Thakur** [1,2], Sanket **Kachole** [1,2], Spyridon **Bakas** [1,2,3,4]

1 Division of Computational Pathology, Department of Pathology and Laboratory Medicine, Indiana University School of Medicine, Indianapolis, IN, USA
2 Indiana University Simon Comprehensive Cancer Center, Indianapolis, IN, USA
3 Departments of Biostatistics and Health Data Science; Radiology and Imaging Sciences; Neurological Surgery, Indiana University School of Medicine, Indianapolis, IN, USA
4 Department of Computer Science, Luddy School of Informatics, Computing, and Engineering, Indiana University, Indianapolis, IN, USA

Abstract

Computational pathology (CompPath) workflows increasingly rely on computational embeddings from digitized hematoxylin and eosin (H&E) whole-slide images (WSIs). Generating such embeddings reproducibly and at scale requires large image transfers, magnification-aware tiling, model-specific preprocessing, versioned software environments, and substantial graphics processing unit (GPU) inference time. Here, we introduce the **T**issue and **R**eport **A**tlas of **C**omputational **E**mbeddings (**TRACE**), as a comprehensive atlas of standardized, multi-scale, and multi-encoder pathology embeddings to eliminate the repetition of this inference burden and expedite CompPath discoveries. TRACE provides multi-magnification (5×, 10×, 20×) histopathology embeddings from diagnostic H&E WSIs, as well as text embeddings from associated diagnostic clinical reports, at the scale of The Cancer Genome Atlas (TCGA) and the Clinical Proteomic Tumor Analysis Consortium (CPTAC) entire data collections. In favor of promoting reproducible reuse of these embeddings and clinically-relevant analyses, TRACE also offers patient-level related clinical (e.g., demographic, treatment), molecular, computational (e.g., preprocessing parameters, tensor metadata), and model provenance information. Sanity-check computational validation of the provided embeddings confirm disease-relevant signal, enabling researchers to move directly to expedited, fair, and reproducible CompPath studies. TRACE is released with its accompanying code and documentation via HuggingFace: https://huggingface.co/datasets/IUCompPath/TRACE

Keywords

Computational pathology, foundation models, whole-slide images, clinical reports, embeddings, multiple-instance learning, TCGA, CPTAC

Article informations

https://doi.org/10.59275/j.melba.2026-c671
Volume 2026, Received: 2026-06-19, Published 2026-09-21
Corresponding author: spbakas@iu.edu
Special issue: MICCAI Open Data 2026 x MELBA
Guest editors: AT, MS, et al.

1. Background

Pathology is the frontline assessment for cancer diagnosis, linking tissue morphology to tumor type, grade, invasion, and clinical decision-making Noller et al. (2025). The digitization of hematoxylin and eosin (H&E) glass slides into Whole-Slide Images (WSIs) has enabled computational analysis of morphology at gigapixel resolution, defining the field of Computational Pathology (CompPath) and enabling to address numerous tasks (e.g., disease classification Innani et al. (2025b,a), prognostication analysis Baheti et al. (2024, 2025), and recurrence prediction You et al. (2025)), using multiple-instance learning (MIL) and multi-modal analyses Kachole et al. (2026). The evolution of pathology to CompPath creates the central opportunity addressed here, since representation learning can accelerate cancer research, but large-scale representation extraction remains expensive to repeat.

A typical WSI-to-embedding workflow requires users to download diagnostic slides from data repositories, such as the Genomic Data Commons (GDC) Grossman et al.

(2016), parse slide metadata, identify tissue regions, extract patches at one or more magnification levels, and run each patch through encoder-specific preprocessing, and GPU inference. Pretrained pathology foundation models (FMs) amplify both the value and the cost of this workflow. FMs such as UNI2-h Chen et al. (2024), Virchow2 Zimmermann et al. (2024), CONCH (**CON**trastive learning from **C**aptions for **H**istopathology) Lu et al. (2024), Prov-GigaPath Xu et al. (2024), and H-Optimus-1 Bioptimus (2024) provide high-dimensional patch representations for Clustering-constrained Attention Multiple-instance learning (CLAM) Lu et al. (2021); Campanella et al. (2019), survival prediction Mobadersany et al. (2018), and multimodal image–text alignment approaches including Pathology Language Image Pre-training (PLIP) Huang et al. (2023) without pixel-level retraining Moor et al. (2023). However, each new encoder, magnification, patch size, or preprocessing policy can require a separate extraction run. At the TCGA/CPTAC scale, this creates repeated terabyte-scale data movement, GPU-hours of inference, and environment-management costs that different research groups would otherwise bear independently Kang et al. (2023); Filiot et al. (2023).

TCGA and CPTAC provide a natural setting for a shared embedding atlas, as they connect diagnostic pathology imaging and report data to clinical and molecular variables under standardized identifiers. TCGA characterized primary cancer types and provides clinical, molecular, biospecimen, radiology, and pathology data Weinstein et al. (2013); Tomczak et al. (2015); Hutter and Zenklusen (2018). CPTAC extends these resources through proteogenomic profiling that links genomic alterations, protein abundance, post-translational regulation, pathology, and clinical phenotypes Edwards et al. (2015); Gillette et al. (2020). Collectively, these data collections have supported many pan-cancer CompPath studies Saltz et al. (2018); Kather et al. (2020); Bulten et al. (2022), but downstream users still need a standardized and reusable representation layer.

Diagnostic reports create a parallel text-representation problem. Such reports contain specimen type, tumor diagnosis, grade, stage, margin status, lymphovascular invasion, ancillary study results, and other clinically relevant details Thirunavukarasu et al. (2023); Liu et al. (2023). However, they appear in heterogeneous institution-specific formats, variable section names, typographic artifacts, and duplicated content. Reproducible text report embedding requires consistent preprocessing and stable linkage to image-derived embeddings Kefeli and Tatonetti (2024).

In this article, we address the common/consistent pathology imaging/report curation and we eliminate the repetitive terabyte-scale data movement and GPU-hours of inference, by releasing the **T**issue and **R**eport **A**tlas of **C**omputational **E**mbeddings (**TRACE**), as a comprehensive atlas of standardized, multi-scale, multi-encoder pathology embeddings. TRACE (v1) provides precomputed patch-level embeddings, slide-level representations, diagnostic report embeddings, and embedding records indexed by repository-specific case, slide, and data collection identifiers. We distinguish *embeddings* from *embedding records*, which describe the units systematically linking WSI embeddings/vectors, with their identifiers, associated clinical, molecular, and computational metadata, all essential to reuse them reproducibly. More specifically, TRACE (v1) spans across 26 organs, comprising 32 cancer types, 41 data collections, >10,000 patients, >16,000 WSIs, and >8,000 reports (Figure 1). By reducing the burden of pathology WSI curation (download, tiling, model-specific preprocessing), GPU inference, and text preprocessing from the CompPath community, TRACE mitigates the repetitive large-scale embedding extraction by independent researchers and expedites computational research and scientific discovery.

2. Summary

TRACE (v.1) mitigates the repetitive computations from the TCGA and CPTAC data collections. Instead of requiring research groups to independently download WSIs, implement tiling, configure multiple FMs, and run GPU inference, we precompute a standardized embedding layer using fixed and recorded software versions, model checkpoints, and model-specific preprocessing settings, and we release the resulting embeddings with their associated metadata. This design enables downstream users to compare encoders, train slide-level models, develop multimodal algorithms, and train CompPath workflows without requiring specialized GPU clusters Lhoest et al. (2021).

TRACE (v.1) considers TCGA and CPTAC data collections with diagnostic WSI coverage, multi-magnification tiling, patch-level embeddings, slide-level representation summaries, and diagnostic report embeddings (Table 1). Vision encoder embeddings are released following quality control, including successful feature extraction, valid tensor shapes, coordinate consistency checks, and metadata/model-registry validation. Report embeddings are provided using 3 text encoders: gemini-embedding-2 Team et al. (2023), Alibaba-NLP/gte-Qwen2-7B-instruct Yang et al. (2024), and google/embe 300m Team et al. (2024).

Each embedding record includes the case ID, slide ID, data collection or study ID, repository, sample type, magnification, encoder name, encoder version, input patch size, feature dimensionality, tensor shape, preprocessing parameters, and quality-control metadata. Patch-level records also include level-0 slide coordinates, tissue-fraction estimates, blur flags, pen/artifact flags when available, and the number of contributing patches used for slide-level aggregation.

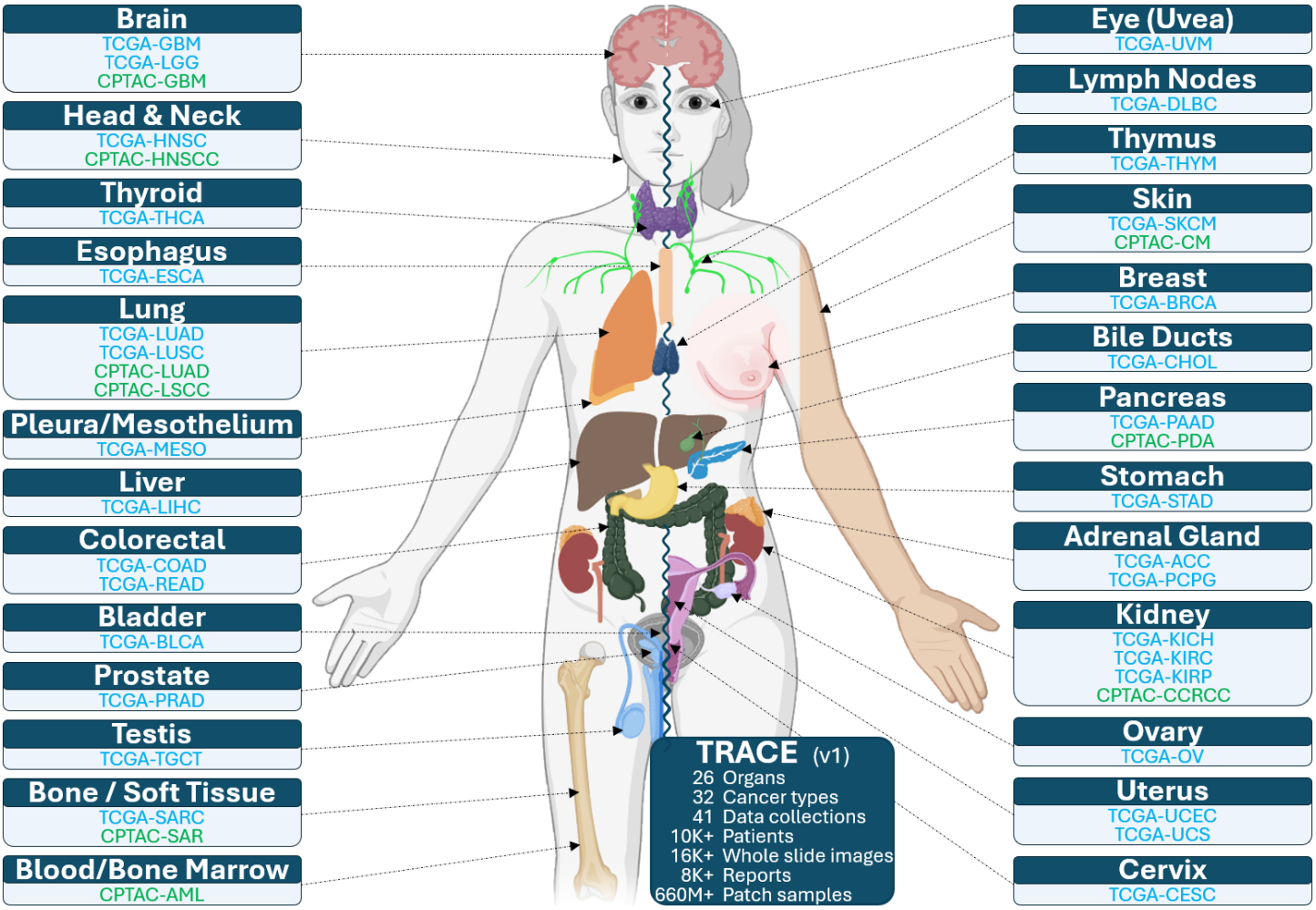


Figure 1: Overview of the **T**issue and **R**eport **A**tlas of **C**omputational **E**mbeddings (**TRACE**). Human anatomical map illustrating the organ systems and listing the TCGA/CPTAC data collections that TRACE (v1) covers. TRACE integrates multim-odal clinical, histopathology, report, and molecular data from 26 organs and >10,000 patients, across 32 cancer types, enabling pan-cancer analyses.

3. Methods

3.1 Cohort description

TRACE includes TCGA and CPTAC data collections with diagnostic H&E WSIs and sufficient slide metadata for magnification-aware tiling. Inclusion required: i) readable slide dimensions and microns-per-pixel (mpp) metadata or a recoverable magnification estimate, ii) acceptance by the versioned tissue-masking and coordinate-generation pipeline; and iii) at least one successful coordinate generation plus at least one successful vision encoder-specific embedding extraction. Slides with missing metadata, unreadable WSIs, insufficient tissue, corrupted coordinate files, failed embedding extraction, or unsupported encoder/patch-size combinations were excluded from embedding tables or retained as failed-status rows in the processing logs. The final data card records the tissue-mask configuration and threshold used to generate the accepted coordinate inventory. Summarized TRACE contents are given in Table 1 and Figure 1, while detailed case counts for each data collection are in Table 3. Cases without available diagnostic text were still included in the image-embedding subset of TRACE. TCGA-LAML was excluded because WSI data were unavailable. The complete data-generation workflow is illustrated in Figure 2.

3.2 Patch-level and slide-level embedding generation

For patch-level encoders, we generated embeddings independently for each valid tissue patch. Patch-level embeddings were stored with their level-0 slide coordinates, enabling users to reconstruct spatial maps, train MIL models, or perform region-level retrieval. Where available, we store the number of contributing patches (N_s), for every slide-level embedding to support downstream weighting and filtering.

Native slide-level embeddings were derived from specialized global slide-level encoders, including CHIEF (Clinical Histopathology Imaging Evaluation Foundation) Wang

Table 1: **TRACE breakdown.** Counts are reported using consistent WSI terminology for both repositories. WSIs are SVS-format files; the coordinate-indexed subset reflects slides with generated coordinate files for a given patch configuration.

Component	Contents
Organs	25 (TCGA) 9 (CPTAC)
Cancer types/Data collections	32 (TCGA) 9 (CPTAC)
Patients	9,548 (TCGA) 1,418 (CPTAC) - all detailed in Suppl.Table 3
Diagnostic WSIs	11,665 (TCGA) 5,258 (CPTAC)
Magnifications	5×, 10×, and 20× where slide metadata and image quality permitted
CPTAC coordinate-indexed subset	5,258 CPTAC slide-coordinate files per CPTAC patch configuration
Vision (patch-level) encoders	UNI, UNI2-h, CONCHv1.5, H-Optimus-1, Phikon-v2, ResNet50, CTransPath, Prov-GigaPath, and Virchow2
Vision (native slide-level) encoders	CHIEF, TITAN, and Prov-GigaPath-Slide
Text embedding models	gemini-embedding-2; Alibaba-NLP/gte-Qwen2-7B-instruct; google/embeddinggemma-300m
Released representation levels	Patch-level embeddings, slide-level representations, report-level embeddings
Feature-record metadata	Case ID, slide ID, repository, data collection/study ID, magnification, x/y coordinates, tissue fraction, QC flags, encoder name, encoder version, input size, embedding dimension, precision, and software environment
Storage format	Parquet-style metadata tables, sharded tensor files, HDF5 coordinate files, and Hugging Face Datasets-compatible manifests
Validation	checksum, coordinate reconstruction, image/text embedding sanity evaluation

et al. (2024), TITAN (Transformer-based pathology Image and Text Alignment Network) Ding et al. (2024), and Prov-GigaPath Xu et al. (2024), and were stored separately from mean-pooled patch-level embeddings. For FMs with native slide-level aggregation, we used the published aggregation method when available and recorded the corresponding preprocessing assumptions. High-dimensional embeddings were generated using the Trident framework (Zhang et al. (2025)), which standardizes feature representation and embedding pipelines across heterogeneous data inputs. All vision encoders are summarized in Table 2.

3.3 Compute accounting

A central motivation for TRACE is to avoid repeated large-scale inference. For each encoder m, we estimate per-patch Floating Point Operations (FLOPs) F_m using operator-level tracing on a single input patch at m's native resolution. We measure inference throughput $T_{m,b}$ in patches per second on NVIDIA A100 at batch size b, after warmup. If TRACE contains N_m encoded patches for model m, the approximate inference burden to regenerate those embeddings is:

$$\text{FLOPs}_m = N_m F_m. \tag{1}$$

The corresponding single-GPU inference time estimate is:

$$\text{GPU-hours}_{m,b} = \frac{N_m}{3600 T_{m,b}}. \tag{2}$$

For each benchmarked patch-level encoder, we report the best measured non-out-of-memory throughput under the recorded benchmark batch-size and precision settings. Slide-level encoders do not consume a lot compute and hence are not reported with their metrics. These estimates quantify the compute avoided by releasing precomputed embeddings. They do not include WSI download time, tissue masking, patch extraction, CPU preprocessing, storage writing, or failed and retried jobs. Thus, they are considered conservative estimates of the resource burden avoided by users.

4. Validation

4.1 Downstream sanity-check tasks

We evaluate the utility of the released patch-level, slide-level, and text-level embeddings using downstream sanity evaluation. These experiments are intended to validate the disease-relevant signal of the embeddings, rather than to establish a definitive benchmark (Table 8).

The available validation artifacts support completed image-only classification for TCGA and CPTAC, as well as text-embedding classification and retrieval-style analy-

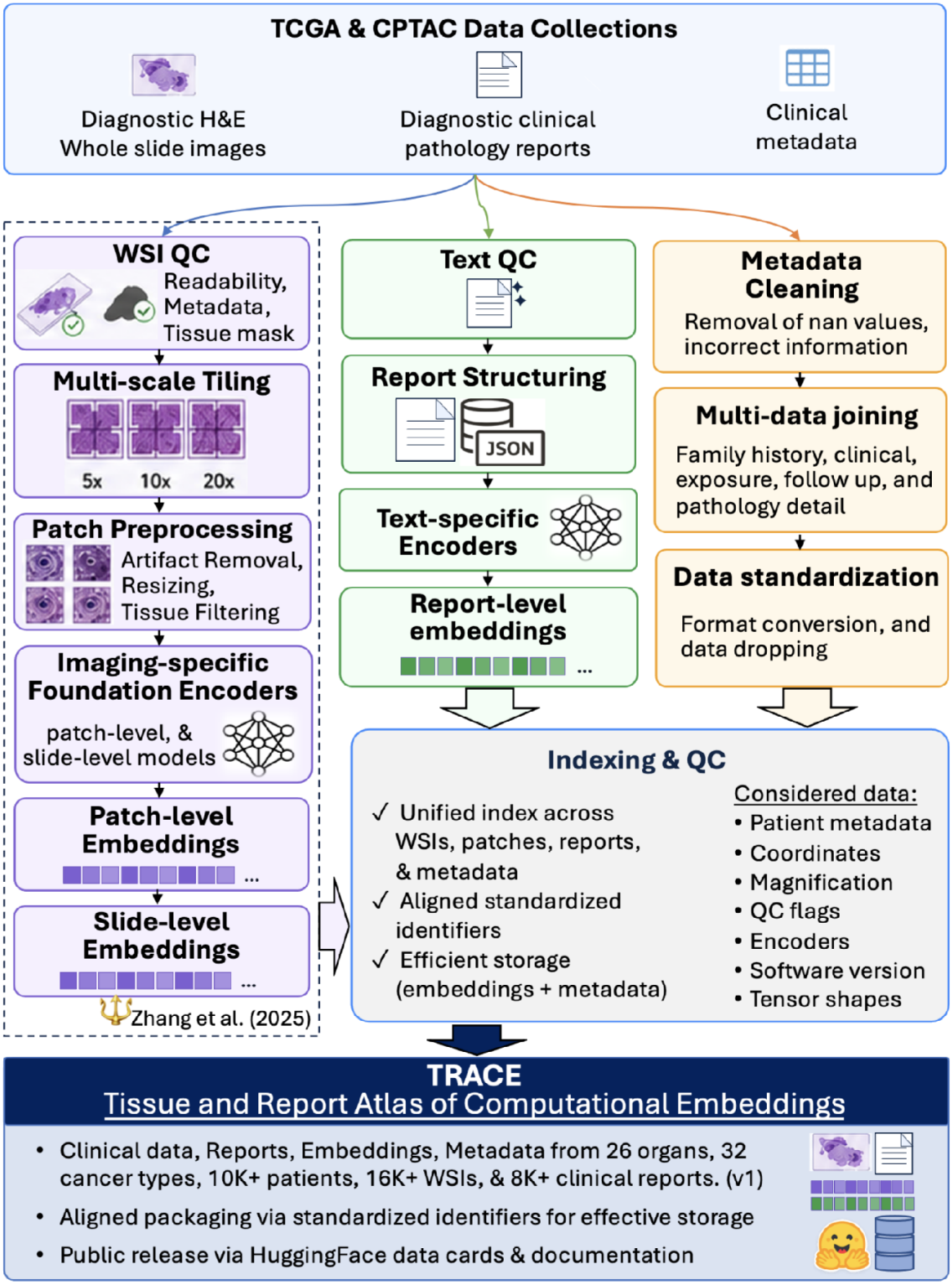


Figure 2: **Flow diagram of the entire pipeline followed to generate the Tissue and Report Atlas of Computational Embeddings (TRACE) (v1).** TCGA and CPTAC diagnostic H&E WSIs and available diagnostic reports are identified and collected, quality controlled, and linked to repository-specific case, slide, and data collection identifiers. Valid WSIs are tessellated at 5×, 10×, and 20× magnification. Tissue patches are processed by pathology imaging-specific vision FMs to produce patch embeddings, which are aggregated into slide-level representations using standardized pooling (mean pooling) or native slide-level FMs. Diagnostic report text is cleaned, optionally structured, and embedded with text models where available. The released artifacts are embedding vectors and embedding records containing provenance, coordinates, quality-control flags, tensor metadata, and model-registry information. Rejected or unsupported inputs are tracked separately from the released artifacts.

Table 2: **Vision encoder specifications.** Values were derived from the local model registry and model-card metadata; patch-level encoders and native slide-level encoders are distinguished in the Type column.

Encoder	Type	Input size	Magnifications	Embedding dim.	License
UNI	Patch-Level	224	5×, 10×, 20×	1,024	CC-BY-NC-ND-4.0
UNI2-h	Patch-Level	224	5×, 10×, 20×	1,536	CC-BY-NC-ND-4.0
CONCHv1.5	Patch-Level	512	5×, 10×, 20×	768	CC-BY-NC-ND-4.0
Virchow2	Patch-Level	224	5×, 10×, 20×	2,560	CC-BY-NC-ND-4.0
Prov-GigaPath	Patch-Level	224	5×, 10×, 20×	1,536	Apache-2.0
H-Optimus-1	Patch-Level	224	5×, 10×, 20×	1,536	CC-BY-NC-ND-4.0
Phikon-v2	Patch-Level	224	5×, 10×, 20×	1,024	Owkin non-commercial license
CTransPath	Patch-Level	224	5×, 10×, 20×	768	GPL-3.0 / non-commercial weights
ResNet50	Patch-Level	256	5×, 10×, 20×	2,048	BSD-3-Clause
CHIEF	Slide-Level	256	10×	768	AGPL-3.0; non-commercial weights
TITAN	Slide-Level	256	20×	768	CC-BY-NC-ND-4.0
Prov-GigaPath	Slide-Level	224	20×	1,536	Apache-2.0

ses for TCGA reports. They do not contain completed multimodal classifier outputs or CPTAC text-report classifier outputs. We report validation results by modality and cohort, distinguishing image-only classification from text-only report-embedding validation.

4.2 Image-feature validation

We first evaluate slide-level image embeddings via a baseline image-only cancer sub-type classification. For TCGA, we trained a 43-class pan-cancer sub-type classifier using UNI-2h embeddings extracted at 20× magnification and evaluated it using a single 70/10/20 training/validation/testing data partitioning. The 43 sub-type classes (Table 6) are different from the 41 data collections and 32 cancer types shown in Figure 2. For CPTAC, we used the same image-only validation strategy for a 9-class classification. The pan-cancer classifiers are distinguishing the different sub-types of cancer types available in Table 3. These experiments provide single-modality image-feature sanity evaluation.

We record the vision encoder registry used for artifact validation in Table 2. We also quantified the inference burden avoided by releasing precomputed patch-level embeddings in Table 7. GPU-hour estimates were computed from final patch counts and measured throughput, while observed logged extraction time from the runtime summary is reported separately in the text.

4.3 Text-feature validation

When diagnostic report text was available, we evaluated whether report embeddings preserved cancer-type signal, by comparing two diagnostic-report preprocessing variants: i) minimally cleaned report text Kefeli and Tatonetti (2024) and ii) structured report text. We evaluated nearest-neighbor retrieval by cancer type, pairwise same-class vs different-class cosine similarity, and linear-probe classification. To avoid leakage, same patient reports were excluded from nearest-neighbor retrieval when identifiers were available, and classification used 5-fold stratified group cross-validation. For TCGA text-only cancer-type classification, logistic regression classifiers trained on cleaned Qwen report embeddings achieved accuracy of 0.941 and macro-F1 of 0.924. The corresponding multiclass classifier macro-AUROC was 0.9876. Separately, we computed a same-class vs different-class cosine AUROC (0.981) as a retrieval and separation diagnostic, rather than as a multiclass classifier AUROC.

Compared with embeddings from minimally cleaned report text, embeddings from structured report text improved same-class vs different-class cosine AUROC from 0.950 to 0.981 and improved macro-F1 from 0.908 to 0.924 relative to minimally cleaned report embeddings. The same-class vs different-class cosine separation gap changed from 0.217 to 0.214. These analyses support the inclusion of standardized diagnostic text embeddings in TRACE, while retaining minimally cleaned text-derived embeddings for users who prefer alternative preprocessing.

4.4 Compute-value validation

To quantify the practical value of precomputation, we benchmarked each patch-level encoder on NVIDIA A100 GPU using identical input resolution, precision, and preprocessing as for the construction of TRACE. The number of patch forward passes varies by encoder and coordinate configuration. Table 7 reports details of N_m per encoder. The per-patch FLOPs ranged from 4.3-312.2 GFLOPs, and measured throughput ranged from 87.0-4,233.2 patches per second. Based on these measured throughputs, recomputing the listed patch embeddings would require an estimated

2,815.4 single-A100 GPU-hours for inference alone.

The observed feature-extraction runtime summary recorded >6,000 logged GPU-hours across 28 encoder/magnification/patch-size runs with elapsed-time data. Because most runtime-summary rows include failed or missing chunks, this value should be interpreted as logged extraction time, rather than as a clean full-regeneration estimate.

5. Resource Availability

5.1 Summary statement

TRACE is a comprehensive atlas of standardized, precomputed, multi-scale, multi-encoder pathology embeddings, with its v1 focusing on the entire TCGA and CPTAC data collections. TRACE pairs diagnostic WSIs embeddings with diagnostic report embeddings, when reports were available. By releasing TRACE via HuggingFace we anticipate a 3-fold contribution: *1)* lower the cost of CompPath research, since repeated slide-level and patch-level inference is unnecessary, *2)* improve reproducibility and fair comparison of relevant CompPath studies, and *3)* expedite scientific discovery through downstream studies of classification, retrieval, weakly supervised learning, multi-modal modeling, and cross-model benchmarking.

5.2 Data and code availability

TRACE URL, incl. documentation and examples:
https://huggingface.co/datasets/IUCompPath/TRACE/
Release version (date): v1.0 (2^{nd} August 2026)

5.3 Licensing

Use of TRACE is subject to the data-use terms of the source repositories and the license terms of the corresponding FMs/encoders (Table 2). Embeddings generated from each FM/encoder should be used in accordance with the model authors' license and terms of use. TCGA WSIs and metadata remain governed by applicable NIH GDC policies. TRACE release includes a model-by-model license table documenting source repositories, weight licenses, citation requirements, and restrictions on commercial or clinical use.

5.4 Usage Considerations

The intention of TRACE is not to be a diagnostic medical device. Users must not attempt re-identification of participants and must comply with TCGA, CPTAC, GDC, and model-specific data-use terms. Because TRACE contains derived representations of human pathology data, users should consider the possibility that embeddings may include information about disease state, tissue source, and cohort membership. Models trained on TRACE should be evaluated for data shift, batch effects, and unintended clinical use before any potential translational application.

6. Discussion

In the article, we presented TRACE, a comprehensive atlas of standardized, precomputed, multi-scale, multi-encoder pathology embeddings, with its v1 focusing on the TCGA and CPTAC data collections. TRACE addresses a practical bottleneck in CompPath: public WSIs and public FMs are available, but extracting embeddings at scale remains expensive, complex, and difficult to reproduce. By releasing a standardized embedding layer with provenance, coordinates, quality-control flags, and text embeddings, TRACE allows researchers to focus on downstream modeling and biological questions rather than repeating large-scale inference. Its key contribution is a reusable representation resource. Many CompPath studies begin by extracting embeddings from WSIs, yet differences in tiling, magnification, masking, encoder versions, pooling strategies, and QC, affect their reproducibility and comparison of their results. TRACE standardizes these steps across multiple encoders and magnifications, and hence supporting cross-model comparison, faster prototyping, and improved reproducibility.

The multi-scale design is important as histologic information spans across magnifications. Low magnification captures tissue architecture, tumor growth pattern, and stromal context. Intermediate magnification captures glandular, papillary, and infiltrative patterns. High magnification captures cellular and nuclear morphology. By providing embeddings at 5×, 10×, and 20×, TRACE allows users to test whether downstream tasks benefit from a single scale, multi-scale concatenation, or learned scale-specific pooling.

The multi-model design is equally important. Pathology FMs differ in architecture, training data, objectives, and input preprocessing. No single encoder is likely to dominate all organs, tumor types, and prediction tasks. By releasing embeddings from multiple encoders under a common cohort and tiling definition, TRACE is expected to enable systematic benchmarking and ensemble strategies for numerous studies. Researchers can compare models without conflating performance with differences in slide preprocessing.

Report embeddings expand TRACE from an image feature resource into a multi-modal representation resource. Reports contain information that may not be visually obvious from a single WSI, including specimen context, tumor grade, margin status, ancillary studies, staging details and even notes from the patient's clinical picture. Paired image and text embeddings enable image-text retrieval, multimodal fusion, report-aware classification, and studies of alignment between morphology and diagnostic language.

Where structured report representations are included, users can evaluate if normalized text improves quality relative to minimally cleaned reports from Kefeli and Tatonetti (2024).

The compute-value analysis shows why precomputed feature release matters. Large-scale feature extraction requires many patch forward passes across multiple encoders and magnifications. Even when each forward pass is fast, the aggregate cost is substantial. TRACE amortizes that cost across the community. Users can load a fixed feature layer and devote compute to downstream modeling rather than repeating the same inference pipeline.

6.1 Limitations

TRACE reduces the barrier to entry for TCGA and CPTAC CompPath studies, but several technical and data-inherent limitations remain. First, frozen representations lock users into the upstream preprocessing decisions used to generate TRACE. Downstream researchers cannot directly evaluate alternative tissue-masking thresholds, stain-normalization algorithms, tile sizes, or overlapping patching strategies from the released embeddings alone. Applications that require end-to-end fine-tuning of vision backbones will still need access to image patches or WSIs.

Second, the provided slide-level summaries impose architectural constraints. Uniform mean pooling over patch embeddings, removes spatial topology and can reduce the contribution of rare diagnostic regions, such as small foci of tumor-infiltrating lymphocytes Baid et al. (2024) or isolated micrometastases. Users who need attention-based MIL, or graph-based pooling should use the patch-level embedding-record and coordinate metadata.

Third, the multi-scale design depends on slide metadata quality. Extracting embeddings at 5×, 10×, and 20× assumes that each WSI contains accurate microns-per-pixel metadata or a reliable fallback magnification estimate. Older files may require heuristic recovery. On the multimodal side, text coverage is incomplete. A single surgical pathology report may also summarize an entire case rather than a specific tissue block, so report embeddings cannot always be spatially aligned to slide-level or patch-level coordinates without introducing cross-slide label noise.

6.2 Future work

Future work will focus on expanding TRACE, by including additional FMs, additional magnifications where appropriate, optional spatial graph embeddings, learned slide-level aggregations, additional report embeddings, and curated benchmark partitions. Community contributions are welcome and will be accepted through the pipeline repository, with requirements for reproducible environments, model license documentation, and validation checks.

A longer-term goal is a living benchmark ecosystem. As new pathology FMs are released, their embeddings can be added to TRACE under the same cohort, tiling, and metadata framework, allowing the community to evaluate whether new FMs improve downstream performance across cancer types, cohorts, magnifications, and multimodal tasks, rather than relying on isolated evaluations.

7. Conclusion

TRACE provides a reusable, precomputed representation layer for CompPath. By combining multi-scale histology embeddings from multiple public vision encoders with diagnostic report embeddings and detailed metadata, TRACE reduces the cost of entry for cancer representation learning. It enables immediate downstream modeling without repeated WSI inference, supports fair cross-model and cross-cohort comparisons, and creates a foundation for multimodal studies that link morphology, text, molecular data, and clinical variables. The resource is designed to accelerate cancer research by making state-of-the-art pathology representations easier to access, reproduce, and extend.

Acknowledgements

Research reported in this publication was partially supported by the Informatics Technology for Cancer Research (ITCR) funding program of the NIH/NCI, under award number U24CA279629. Computational resources were partially supported by Lilly Endowment, Inc., through the Indiana University Pervasive Technology Institute. The authors would like to thank Hugging Face for sponsoring the long term data storage and facilitating the dissemination of this resource. The authors would also like to acknowledge the contribution of Varalakshmi Perumal on designing Figure 1.

Conflict of Interest

The authors declare no conflicts of interest.

Ethical Standards

No IRB approval was required because this study uses secondary, de-identified derived publicly available data.

Author Contributions

ST, SB conceived the study. ST developed the embedding extraction pipeline. ST & SK curated the TCGA & CPTAC data, respectively. ST implemented vision model inference. ST & SK implemented text preprocessing. ST performed validation experiments, prepared the data card, release documentation, and wrote the initial manuscript. All authors reviewed and approved the final manuscript.

Data/Code Availability

Use of the released data is subject to the policies of the source repositories and the licenses of the corresponding FMs. TRACE, including feature files, metadata tables, schema documentation, and data cards, is available at https://huggingface.co/datasets/IUCompPath/TRACE. Pipeline code, model registry, Docker configuration, validation scripts, & example notebooks for loading TRACE, training linear probes, running experiments, reconstructing coordinate heatmaps, & reproducing the compute accounting are at huggingface.co/datasets/IUCompPath/TRACE/examples.

References

C. Abbet et al. Divide and rule: Self-supervised learning for survival analysis. In *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 480–489. Springer, 2020.

G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (ICLR)*, 2017.

B. Baheti et al. Prognostic stratification of glioblastoma patients by unsupervised clustering of morphology patterns on whole slide images furthering our disease understanding. *Frontiers in Neuroscience*, 18, May 2024. ISSN 1662-453X. . URL http://dx.doi.org/10.3389/fnins.2024.1304191.

B. Baheti et al. Multimodal explainable artificial intelligence for prognostic stratification of patients with glioblastoma. *Modern Pathology*, 38(9):100797, 2025. ISSN 0893-3952. . URL http://dx.doi.org/10.1016/j.modpat.2025.100797.

U. Baid et al. Pan-cancer tumor infiltrating lymphocyte detection based on federated learning. In *2024 IEEE International Conference on Big Data (BigData)*, page 7640–7647. IEEE, Dec. 2024. . URL http://dx.doi.org/10.1109/BigData62323.2024.10825083.

S. Bakas et al. Brats-path challenge: Assessing heterogeneous histopathologic brain tumor sub-regions, 2024. URL https://arxiv.org/abs/2405.10871.

A. Bardes, J. Ponce, and Y. LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022.

B. E. Bejnordi et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*, 318(22):2199–2210, 2017. .

Bioptimus. H-optimus-0. Hugging Face model, 2024. URL https://huggingface.co/bioptimus/H-optimus-0.

W. Bulten et al. Artificial intelligence for diagnosis and gleason grading of prostate cancer: the panda challenge. *Nature Medicine*, 28(1):154–163, 2022. .

G. Campanella et al. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, 25(8):1301–1309, 2019. .

M. Caron et al. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9650–9660, 2021.

R. J. Chen et al. Multimodal co-attention transformer for survival prediction in gigapixel whole slide images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4025, 2021.

R. J. Chen et al. Towards a general-purpose foundation model for computational pathology. *Nature Medicine*, 30(3):850–862, 2024. .

T. Chen et al. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607. PMLR, 2020.

T. Ding et al. Multimodal whole slide foundation model for pathology. *arXiv preprint arXiv:2411.19666*, 2024.

N. J. Edwards et al. The cptac data portal: A resource for cancer proteomics research. *Journal of Proteome Research*, 14(6):2707–2713, 2015. .

A. Filiot et al. Scaling self-supervised learning for histopathology with masked image modeling. *medRxiv*, 2023. .

M. A. Gillette et al. Proteogenomic characterization reveals therapeutic vulnerabilities in lung adenocarcinoma. *Cell*, 182(1):200–225, 2020. .

R. L. Grossman et al. Toward a shared vision for cancer genomic data. *New England Journal of Medicine*, 375 (12):1109–1112, 2016. .

K. He et al. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9729–9738, 2020.

Z. Huang et al. A visual-language foundation model for pathology image analysis using medical twitter. *Nature Medicine*, 29(9):2307–2316, 2023. .

C. Hutter and J. C. Zenklusen. The landscape of genomics in cancer. *Cell*, 173(2):305–320, 2018. .

M. Ilse, J. Tomczak, and M. Welling. Attention-based deep multiple instance learning. In *International Conference on Machine Learning (ICML)*, pages 2127–2136. PMLR, 2018.

S. Innani et al. Interpretable artificial intelligence based determination of glioma idh mutation status directly from histology slides. *Neuro-Oncology Advances*, 7(1): vdaf140, 2025a.

S. Innani et al. Ai-driven who 2021 classification of gliomas based only on h&e-stained slides. *Neuro-Oncology*, 28 (1):282–296, Aug. 2025b. ISSN 1523-5866. . URL http://dx.doi.org/10.1093/neuonc/noaf189.

S. Kachole et al. Multi-frugal: Multimodal flexible redundancy-aware decomposed gated learning for cancer diagnosis and prognosis, 2026. URL https://arxiv.org/abs/2606.06867.

M. Kang et al. Benchmarking self-supervised learning methods for digital pathology. *arXiv preprint arXiv:2304.08078*, 2023.

J. N. Kather et al. Pan-cancer image-based detection of clinically actionable genetic alterations. *Nature Cancer*, 1(8):789–799, 2020. .

J. Kefeli and N. P. Tatonetti. Tcga-reports: A machine-readable pathology report resource for benchmarking text-based ai models. *Patterns*, 5(3):100933, 2024. .

Q. Lhoest et al. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, 2021.

H. Liu et al. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

M. Y. Lu et al. Data-efficient and weakly supervised computational pathology on whole-slide images. *Nature Biomedical Engineering*, 5(6):555–570, 2021. .

M. Y. Lu et al. A visual-language foundation model for computational pathology. *Nature Medicine*, 30(3):863–874, 2024. .

S. M. McKinney et al. International evaluation of an ai system for breast cancer screening. *Nature*, 577(7788): 89–94, 2020. .

P. Mobadersany et al. Predicting cancer outcomes from histology and genomics using convolutional networks. *Proceedings of the National Academy of Sciences (PNAS)*, 115(13):E2970–E2979, 2018. .

M. Moor et al. Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265, 2023. .

K. Noller et al. Informatics at the frontier of cancer research. *Cancer Research*, 85(16):2967–2986, 2025. ISSN 1538-7445. .

M. Oquab et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

N. Reimers and I. Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3982–3992, 2019.

J. Saltz et al. Spatial organization and molecular correlation of tumor-infiltrating lymphocytes using deep learning on pathology images. *Cell Reports*, 23(1):181–193, 2018. .

G. Team et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

G. Team et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

A. J. Thirunavukarasu et al. Large language models in medicine. *Nature Medicine*, 29(8):1930–1940, 2023. .

K. Tomczak, P. Czerwińska, and M. Wiznerowicz. The cancer genome atlas (tcga): an immeasurable source of knowledge. *Contemporary Oncology*, 19(1A):A68–A77, 2015. .

A. Vaswani et al. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 5998–6008, 2017.

X. Wang et al. A pathology foundation model for cancer diagnosis and prognosis prediction. *Nature*, 634:970–978, 2024. .

J. N. Weinstein et al. The cancer genome atlas pan-cancer analysis project. *Nature Genetics*, 45(10):1113–1120, 2013. .

H. Xu et al. A whole-slide foundation model for digital pathology from real-world data. *Nature*, 630:181–188, 2024. .

A. Yang et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

S. You et al. Profuseme: Prostate cancer biochemical recurrence prediction via fused multi-modal embeddings, 2025. URL https://arxiv.org/abs/2509.14051.

A. Zhang et al. Accelerating data processing and benchmarking of ai models for pathology. *arXiv preprint arXiv:2502.06750*, 2025.

E. Zimmermann et al. Virchow2: Scaling self-supervised mixed magnification models in pathology. *arXiv preprint arXiv:2408.00738*, 2024.

Appendix A. Supplementary

A.1 Release metadata

Release name: TRACE
Release version: v.1
Release date: 02 August 2026
GDC access date: 2026-06-05 RAW-DATA snapshot
TCGA data collections included: 32
TCGA cases: 9,548
TCGA diagnostic WSIs: 11,665
CPTAC data collections included: 9
CPTAC cases: 1,418
CPTAC diagnostic WSIs: 5,258
CPTAC coordinate-indexed slide files per patch configuration: 5,258
CPTAC clinical cases in RAW-DATA: 2,025
CPTAC biospecimen slide records in RAW-DATA: 2,432
Number of TCGA reports with text: 8,383

A.2 Model registry

For each model, TRACE records:

- model name;
- model family;
- source repository;
- checkpoint URL or access instructions;
- license;
- version, tag, or commit hash;
- input resolution;
- preprocessing transform;
- embedding dimension;
- inference precision;
- batch size used for atlas extraction;
- hardware used;
- checksum of generated feature shards.

A.3 Target audience

TRACE is designed for four primary audiences. First, CompPath laboratories can use the released embeddings to train and evaluate slide-level models without repeating WSI feature extraction. Second, benchmark organizers can use TRACE as a common representation layer for fair comparison across downstream algorithms Campanella et al. (2019); Bejnordi et al. (2017); McKinney et al. (2020); Bakas et al. (2024). Third, multi-modal learning researchers can join WSI embeddings with report embeddings, molecular features, and clinical variables using stable TCGA identifiers. Fourth, educators can use TRACE to teach weakly supervised learning, retrieval, and representation analysis without requiring specialized GPU infrastructure.

A.4 Readiness

TRACE is distributed in an analysis-ready format through Hugging Face Datasets (Lhoest et al. (2021)). Embeddings can be loaded with standard Python tools. The release includes data cards, schema documentation, model license metadata, fixed example partitions, and basic notebooks for classification with MIL. Basic Python knowledge is sufficient to load the slide-level embeddings and train linear probes or shallow classifiers (Alain and Bengio (2017)).

Appendix B. Method Details for processing pipelines

B.1 WSI processing and tiling

WSI processing was performed using a containerized pipeline; the local benchmark environment recorded Python v.3.12.11, PyTorch v.2.10.0+cu128, and CUDA 12.8, while the OpenSlide v.4.0.1. All software versions and container hashes are recorded in the release data card. For each WSI, we generated a low-resolution thumbnail and estimated tissue masks using HSV thresholding followed by morphological closing. WSIs were tiled at 5×, 10×, and 20× magnification. Patch extraction used non-overlapping tiles unless otherwise specified. Patch sizes were chosen to match each encoder's native input resolution when possible; for models requiring fixed input dimensions, patches were resized according to the model-specific preprocessing transform recorded in the registry; model cards commonly specify bicubic resize/crop for transformer encoders, but a single global interpolation setting was not recorded in the local artifacts.

For every extracted patch, we store the slide ID, TCGA case ID, data collection ID, magnification, level-0 x-coordinate, level-0 y-coordinate, patch width, patch height, tissue fraction, and quality-control flags. Quality-control flags include insufficient tissue fraction, blur based on Laplacian variance, likely pen or marker artifact when detected, and read errors. The local coordinate files record accepted patch coordinates, but tissue-fraction, Laplacian-variance, and artifact-fraction thresholds are not present in the provided artifact bundle. Users should treat the coordinate inventory as the accepted-patch set and apply any stricter QC filters from the final data card when available.

B.2 Vision foundation models

We selected public pathology vision encoders to cover multiple architectures, training datasets, input resolutions, and representation dimensions. The release includes the encoders that passed feature-generation and quality-control checks in the final model registry. For models with restricted access or non-standard licenses, the data card records the exact access conditions and license terms.

Each encoder was run in evaluation mode using the preprocessing pipeline recommended by its authors when available. Input normalization, color-space assumptions, patch size, interpolation mode, precision, and pooling strategy Caron et al. (2021); Oquab et al. (2023); Chen et al. (2020); He et al. (2020); Bardes et al. (2022) are recorded for each model. For transformer-based patch encoders, the released embedding uses each model's recommended feature output, most commonly the classification (CLS) token or the model-card-specified pooled tile embedding; Virchow2 uses the concatenated class token and mean patch-token embedding. For convolutional encoders, the released embedding corresponds to the pooled penultimate feature representation. All model weights were pinned by version, commit hash, release tag, or checkpoint checksum.

B.3 Diagnostic text processing

We obtained diagnostic pathology reports from a cleaned dataset by Kefeli and Tatonetti (2024) and linked them to TCGA cases using the TCGA case ID extracted from the patient filename. For each available report, we generated two text representations. First, we provided the original cleaned report from the dataset. Second, when applicable, we created a structured version by removing non-diagnostic headers and footers, normalizing whitespace, and correcting encoding artifacts. This version was designed to normalize section order while preserving clinical meaning, including fields such as clinical history, specimen submitted, final diagnosis, tumor type, grade, stage, margin status, lymphovascular invasion, lymph node status, ancillary studies, and additional comments.

Missing or unavailable fields were explicitly marked as unavailable rather than inferred. The original report text, minimally cleaned text Kefeli and Tatonetti (2024), and structured text are tracked separately in the metadata where redistribution is permitted. If source-text redistribution is restricted, only derived embeddings and non-sensitive metadata are released. The exact prompt or rules used for structuring are included in the repository to support reproducibility.

B.4 Text embedding models

Report-level embeddings were generated using "gemini-embedding-2" Team et al. (2023), "Alibaba-NLP/gte-Qwen2-7B-instruct" Yang et al. (2024), and "google/embeddinggemma-300m" Team et al. (2024). For each text encoder, we record the tokenizer, maximum sequence length, truncation strategy, pooling method via standard SentenceTransformer mechanisms Reimers and Gurevych (2019); Vaswani et al. (2017), embedding dimension, model version, and software environment.

B.5 Storage format and schema

The atlas is organized around stable repository identifiers. Metadata tables are stored in Parquet format. Each row in the slide-level table corresponds to a unique combination of case ID, slide ID, magnification, encoder, and feature type. Each row in the patch-level table corresponds to a unique patch with coordinates, quality-control flags, and a pointer to the corresponding feature tensor. Embeddings are stored in sharded binary tensor files rather than in the metadata tables. Sharding is organized to support streaming access by data collection or study.

The minimal slide-level schema contains case ID, slide ID, data collection ID, sample type, magnification, encoder, encoder version, feature path, feature dimensions, num patches, and quality control (qc) status.

Appendix C. Quality Control

C.1 Feature integrity

We performed local artifact-integrity checks on the available manifests and generated outputs. Tensor shapes and metadata dimensions were validated against the model registry for every encoder in Table 2, and coordinate inventories were cross-checked against patch-count summaries. Recompute-based numerical reproducibility values were not present in the local artifact bundle, so mean and maximum absolute feature differences are not reported in this manuscript snapshot.

C.2 Coordinate consistency

To verify that patch coordinates aligned with source slides, we reconstructed low-resolution spatial maps from patch-level metadata. For each sampled slide, patch coordinates were projected onto a thumbnail, and tissue coverage was compared with the original tissue mask. The local coordinate inventory contained accepted coordinate files and patch counts, but no separate corrected-slide or systematic-shift log was present. Slides or jobs with coordinate failures are represented through failed or missing runtime/status rows rather than a curated correction table.

C.3 Visual quality control

We generated representative coordinate and compute summaries from patch-level metadata. These summaries were inspected to confirm that patch counts, magnification-specific inventories, and throughput measurements were consistent with the released feature tables. Examples are shown in Figures 3 and 4. The purpose of this validation was not to assign diagnostic saliency but to verify the spatial and computational integrity of the released patch feature table.

Appendix D. Design Rationale

D.1 Potential use cases

The atlas facilitates a diverse array of CompPath workflows across TCGA and CPTAC data collections, ranging from pan-cancer classification using frozen slide embeddings to cancer sub-type categorization via linear probes or MIL Ilse et al. (2018); Lu et al. (2021). It enables biomarker prediction and molecular subtyping by integrating labels from TCGA or derived curated datasets. Furthermore, the atlas serves as infrastructure for cross-model benchmarking, model ensembling across encoders and magnifications Abbet et al. (2020), image retrieval Huang et al. (2023), nearest-neighbor case discovery Filiot et al. (2023), and multi-modal fusion of histology, diagnostic text, and multi-omics arrays Chen et al. (2021). Beyond these applications, the atlas supports image-text retrieval using paired histology and report embeddings, aids reproducibility studies through a fixed feature layer, and provides educational resources for exploring weakly supervised learning, feature visualization, and pathology representation analysis.

Appendix E. Usage Notes

The atlas is intended for downstream research on frozen representations. Users interested in slide-level classification, retrieval, or survival modeling can begin with the slide-level feature table. Users interested in MIL, spatial pooling, or heatmap generation should use the patch-level feature table and coordinate metadata. Because different encoders have different licenses and preprocessing assumptions, users should select embedding subsets compatible with their intended use.

We recommend that downstream studies partition data at the patient level rather than the slide or patch level. When multiple slides are available for a patient, all slides from that patient should remain in the same partition. For text analyses, reports from the same patient should not appear in both retrieval queries and candidate sets unless the task explicitly permits patient-level matching.

Users should also be aware that TCGA and CPTAC data collections are heterogeneous in nature. Slides were generated across many institutions, scanners, staining protocols, tissue-processing workflows, tumor types, and eras. This heterogeneity is a strength for representation analyses, but may also introduce confounding. Downstream studies should consider data collection, tissue site, sample type, and batch effects when interpreting model performance.

E.1 Compute summary

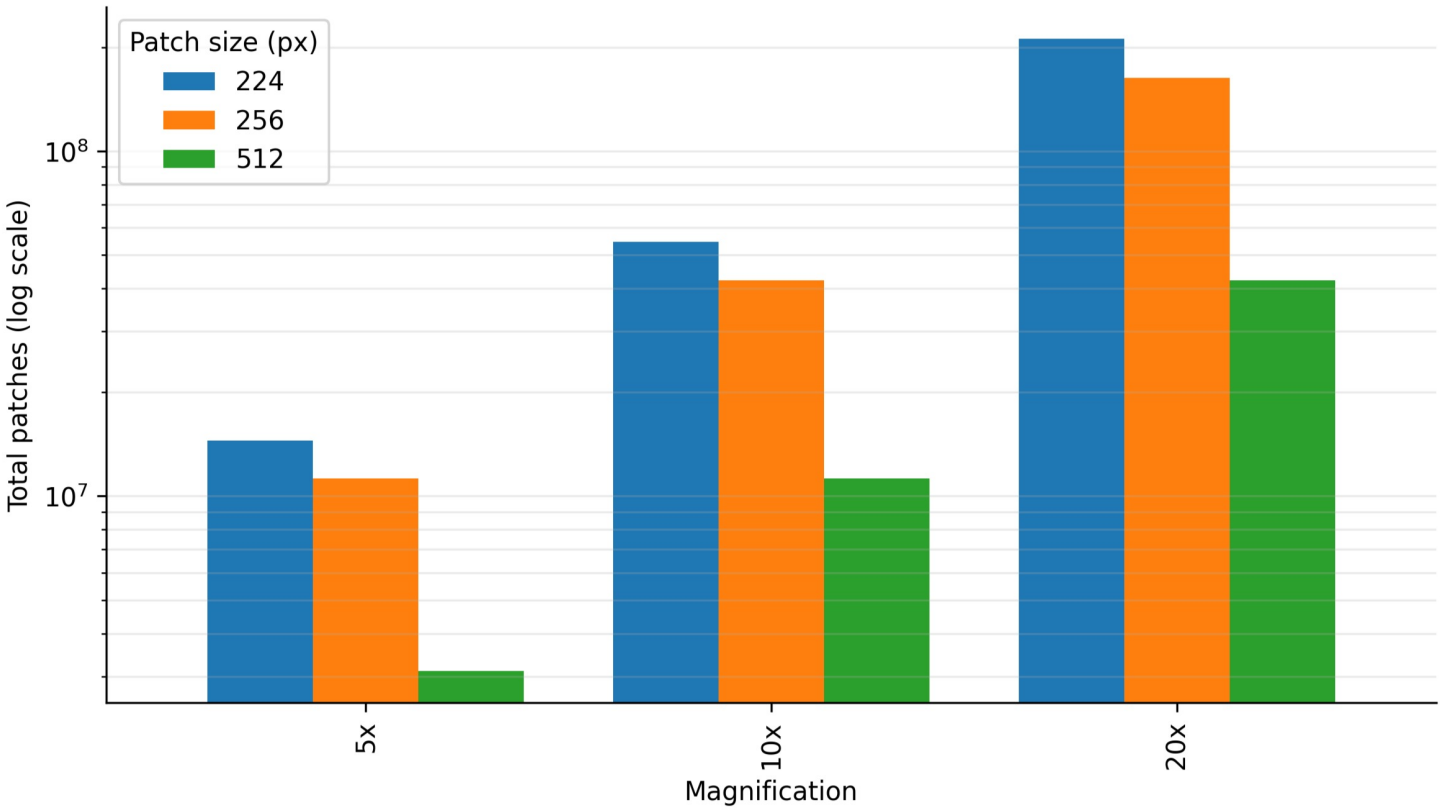
The release required 6000+ logged GPU-hours for vision feature extraction; total CPU-hours for WSI preprocessing, metadata construction, and validation were not present in the local runtime artifacts. Based on measured NVIDIA A100 throughput, regenerating the benchmarked patch-encoder rows in Table 7 would require approximately 2,514.3 single-GPU hours under ideal conditions. The appendix storage/runtime rows use runtime-log scaling where available and benchmark throughput for rows without compatible runtime logs; those displayed row-level estimates sum to approximately 2,514.3 GPU-hours. The logged runtime estimate excludes rows without elapsed runtime and should be interpreted separately from the benchmark-derived regeneration estimate.

Appendix F. Example Loading Code

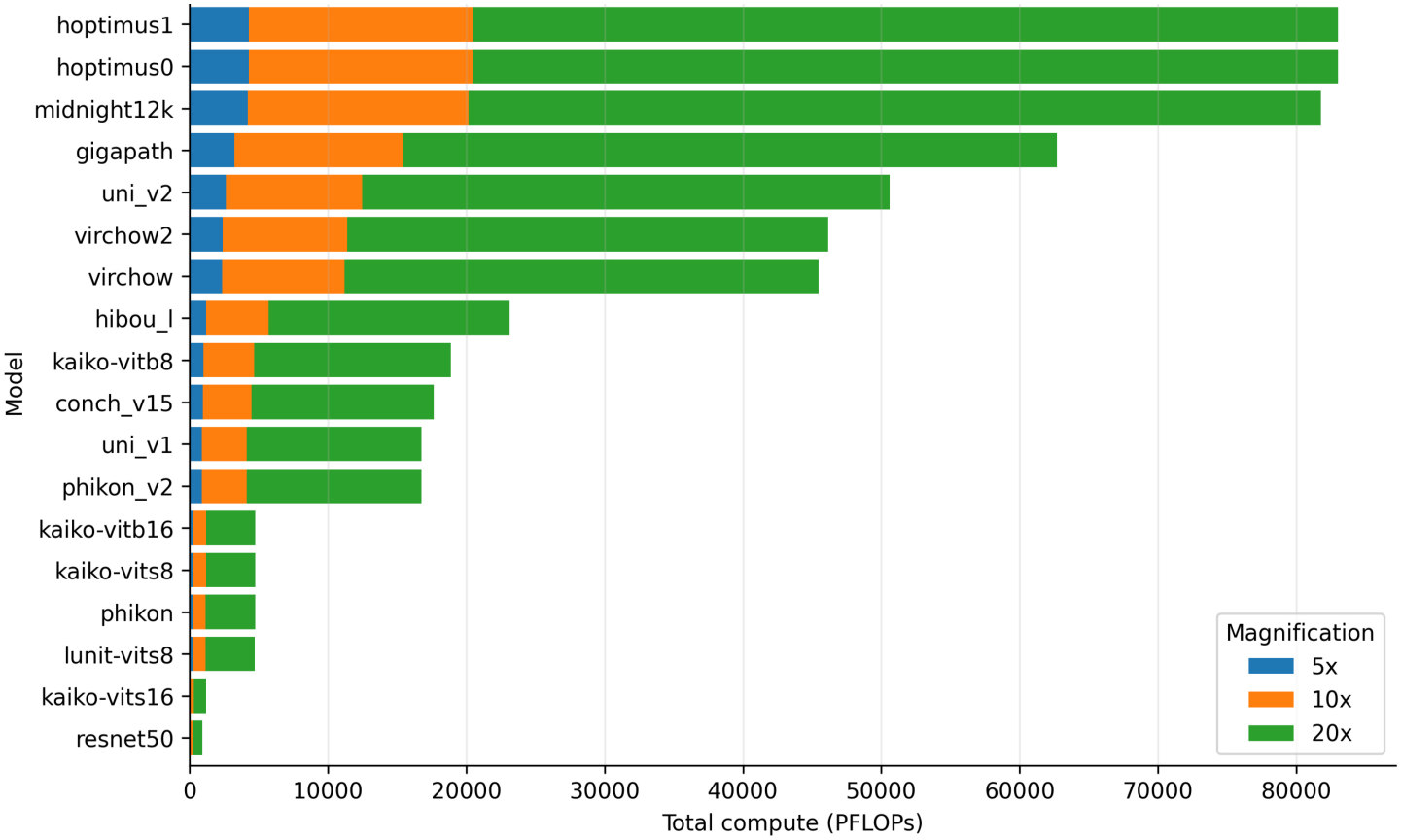
```
from datasets import load_dataset

dataset = load_dataset(
    "IUCompPath/TRACE",
    name="slide_embeddings",
)

row = dataset[0]
print(row.keys())
print(
    row["case_id"], row["slide_id"],
    row["encoder"]
)
```

Appendix G. Recommended Patient-Level Evaluation Protocol

For classification and retrieval tasks, we recommend patient-level partitioning. All WSIs and reports from the same TCGA case should remain in the same partition. For cross-validation, users should use stratified group folds when labels are imbalanced and multiple slides per case are present. Patch-level partitions should be avoided because they leak slide- and patient-specific information.



(a) Dataset Patch Count by Magnification and Patch Size



(b) Compute (PFLOPs) per Encoder For the Complete 650M Patches of TRACE

Figure 3: **Vision coordinate and compute inventory.** (a) Dataset patch count inventory by magnification and patch size from the generated coordinate summaries. (b) Estimated total feature-extraction compute by vision encoder. The listed encoders extend beyond the ones covered in TRACE.

Table 3: Summary of combined TCGA and CPTAC data with data collection counts, reports, and patch totals.

Data Collection	Cases	WSIs / SVS files	Reports with text	Total valid patches
TCGA Data Collections				
TCGA-ACC	56	227	54	14,217,849
TCGA-BLCA	386	457	353	28,223,269
TCGA-BRCA	1,055	1,126	992	55,413,192
TCGA-CESC	269	279	251	10,220,633
TCGA-CHOL	38	38	36	2,549,388
TCGA-COAD	434	442	395	15,113,534
TCGA-DLBC	44	44	43	1,601,263
TCGA-ESCA	156	158	117	6,827,303
TCGA-GBM	389	860	245	29,942,903
TCGA-LGG	491	844	444	31,493,549
TCGA-HNSC	450	472	442	18,714,624
TCGA-KICH	109	121	108	6,247,014
TCGA-KIRC	513	519	501	26,753,372
TCGA-KIRP	272	296	261	13,862,426
TCGA-LIHC	364	372	328	18,166,804
TCGA-LUAD	468	531	438	22,949,432
TCGA-LUSC	478	512	442	22,800,878
TCGA-MESO	75	87	67	3,243,498
TCGA-OV	106	107	62	5,794,064
TCGA-PAAD	183	209	174	10,043,193
TCGA-PCPG	176	196	171	10,868,504
TCGA-PRAD	403	449	368	21,053,699
TCGA-READ	157	158	149	4,186,521
TCGA-SARC	254	600	243	37,062,297
TCGA-SKCM	433	475	97	25,547,370
TCGA-STAD	375	400	296	17,096,332
TCGA-TGCT	149	254	87	14,333,039
TCGA-THCA	506	519	486	25,693,879
TCGA-THYM	121	180	113	10,877,563
TCGA-UCEC	505	566	503	33,977,626
TCGA-UCS	53	87	52	5,334,690
TCGA-UVM	80	80	65	3,116,315
TCGA Subtotal	9,548	11,665	8,383	553,326,023
CPTAC Data Collections				
CPTAC-AML_v3	89	122	–	8,047,619
CPTAC-CCRCC_v1	222	783	–	14,148,369
CPTAC-CM_v7	95	411	–	7,216,079
CPTAC-GBM_v15	178	462	–	5,002,921
CPTAC-HNSCC_v10	112	390	–	6,005,324
CPTAC-LSCC_v10	212	1,081	–	26,246,539
CPTAC-LUAD_v12	244	1,137	–	31,693,812
CPTAC-PDA_v8	178	567	–	5,430,458
CPTAC-SAR_v10	88	305	–	5,491,499
CPTAC Subtotal	1,418	5,258	–	109,282,620
Grand Total	**10,966**	**16,923**	**8,383**	**662,608,643**

Table 4: **Text model specifications.**

Model	Max tokens	Feature dim.
gemini-embedding-2	8,192	3,072
Alibaba-NLP/gte-Qwen2-7B-instruct	32,768	3,584
google/embeddinggemma-300m	8,192	768

Appendix H. Citation and Use Requirements

Users of TRACE should cite this manuscript. They should also cite according to the data they use TCGA, CPTAC, GDC, the source WSI repository, and each FM used in their analysis. A model-by-model citation table is included in the data card.

Appendix I. Diagnostic Text Processing Prompt

If structured diagnostic reports are included, the exact prompt or rule set should be reported here.

```
You are converting OCR-derived surgical pathology r

Task rules:
- Correct obvious OCR and spelling errors.
- Preserve all clinically meaningful facts and all
- Do not infer missing diagnoses, stages, grades, m
- Remove administrative boilerplate, attestations,
- Normalize section headers and whitespace.
- Prefer concise clinical plain text, not JSON.
- Use stable labels and key-value lines.
- If a value is missing, write "not stated".
- Do not add explanations outside the cleaned repor

Return this structure:
Report Type:
Clinical Diagnosis and History:
Specimen Submitted:
Final Diagnosis:
Key Tumor Features:
Gross Description:
Intraoperative Consultation:
Embedding Summary:
Uncertain or Ambiguous Text:
```

Table 5: **Compute and storage summary by encoder and patch configuration.** Patch counts and storage include TCGA and CPTAC data collections. GPU-hours are atlas-wide estimates: rows with completed TCGA logs scale logged TCGA runtime by the TCGA+CPTAC patch ratio; rows without same-configuration logs use model-level logged patch throughput or benchmark throughput.

Encoder	Magnification	Patches	Storage (TCGA + CPTAC)	GPU-hours
UNI2-h	5×	14,470,202	66 GB + 14 GB = 80 GB	19.8
UNI2-h	10×	65,845,834	247 GB + 50 GB = 297 GB	90.2
UNI2-h	20×	253,981,413	948 GB + 191 GB = 1.14 TB	348.1
CONCHv1.5	5×	3,805,530	9.5 GB + 2.1 GB = 12 GB	1.0
CONCHv1.5	10×	13,606,655	34 GB + 7.2 GB = 41 GB	3.5
CONCHv1.5	20×	50,848,702	126 GB + 26 GB = 152 GB	13.1
TITAN	20×	50,848,702	40 MB + 8 MB = 48 MB	8.1
CTransPath	5×	13,617,508	34 GB + 7.2 GB = 41 GB	1.2
CTransPath	10×	50,895,104	132 GB + 27 GB = 159 GB	4.6
CTransPath	20×	195,537,695	483 GB + 97 GB = 580 GB	17.7
CHIEF	10×	50,895,104	40 MB + 8 MB = 48 MB	42.9
Prov-GigaPath	5×	13,617,508	66 GB + 14 GB = 80 GB	112.8
Prov-GigaPath	10×	50,895,104	247 GB + 51 GB = 298 GB	378.2
Prov-GigaPath	20×	195,537,695	947 GB + 191 GB = 1.14 TB	135.9
Prov-GigaPath-Slide	20×	–	90 MB + 18 MB = 108 MB	–
H-Optimus-1	5×	14,470,202	85 GB + 18 GB = 103 GB	33.4
H-Optimus-1	10×	65,845,834	319 GB + 65 GB = 384 GB	8.2
H-Optimus-1	20×	253,981,413	1.30 TB + 262 GB = 1.56 TB	410.2
ResNet50	5×	13,617,508	56 GB + 12 GB = 68 GB	0.9
ResNet50	10×	50,895,104	209 GB + 43 GB = 251 GB	3.3
ResNet50	20×	195,537,695	801 GB + 162 GB = 962 GB	12.8
Virchow2	5×	14,470,202	140 GB + 29 GB = 169 GB	27.1
Virchow2	10×	65,845,834	528 GB + 108 GB = 636 GB	66.6
Virchow2	20×	253,981,413	2.00 TB + 403 GB = 2.40 TB	247.9

Table 6: **Proposed 43-class TCGA-OncoTree taxonomy grouped by anatomic site.** The 46 TCGA-OT labels are reduced to 43 by harmonizing two grade-related glioma label pairs and the overlapping melanoma labels.

Organ/site	Diagnostic subclasses	Organ/site	Diagnostic subclasses
Adrenal gland	ACC: Adrenocortical carcinoma PHC: Pheochromocytoma	**Lung**	LUAD: Lung adenocarcinoma LUSC: Lung squamous cell carcinoma
Bladder	BLCA: Bladder urothelial carcinoma	**Ovary**	HGSOC: High-grade serous ovarian carcinoma
Brain/CNS	ASTR/AASTR: Astrocytoma OAST/AOAST: Oligoastrocytoma ODG: Oligodendroglioma GBM: Glioblastoma	**Pancreas**	PAAD: Pancreatic adenocarcinoma
Breast	IDC: Invasive ductal carcinoma ILC: Invasive lobular carcinoma	**Pleura**	PLEMESO: Pleural mesothelioma
Cervix	CESC: Cervical squamous cell carcinoma	**Prostate**	PRAD: Prostate adenocarcinoma
Colon/rectum	COAD: Colon adenocarcinoma MACR: Mucinous colorectal adenocarcinoma READ: Rectal adenocarcinoma	**Skin**	MEL/SKCM: Melanoma
Esophagus/ stomach	ESCA: Esophageal adenocarcinoma ESCC: Esophageal squamous cell carcinoma DSTAD: Diffuse-type stomach adenocarcinoma STAD: Stomach adenocarcinoma TSTAD: Tubular stomach adenocarcinoma	**Soft tissue**	DDLS: Dedifferentiated liposarcoma LMS: Leiomyosarcoma MFH: Undifferentiated pleomorphic sarcoma MFS: Myxofibrosarcoma
Eye	UM: Uveal melanoma	**Testis**	NSGCT: Non-seminomatous germ cell tumor SEM: Seminoma
Head and neck	HNSC: Head and neck squamous cell carcinoma	**Thymus**	THYM: Thymoma
Kidney	CCRCC: Clear cell renal cell carcinoma CHRCC: Chromophobe renal cell carcinoma PRCC: Papillary renal cell carcinoma	**Thyroid**	THFO: Follicular thyroid carcinoma THPA: Papillary thyroid carcinoma
Liver	HCC: Hepatocellular carcinoma	**Uterus**	UCS: Uterine carcinosarcoma UEC: Uterine endometrioid carcinoma USC: Uterine serous carcinoma

Harmonization note. ASTR/AASTR combines astrocytoma and anaplastic astrocytoma; OAST/AOAST combines oligoastrocytoma and anaplastic oligoastrocytoma; and MEL/SKCM combines the generic melanoma and cutaneous melanoma labels. These combined identifiers are harmonized labels rather than official individual OncoTree codes.

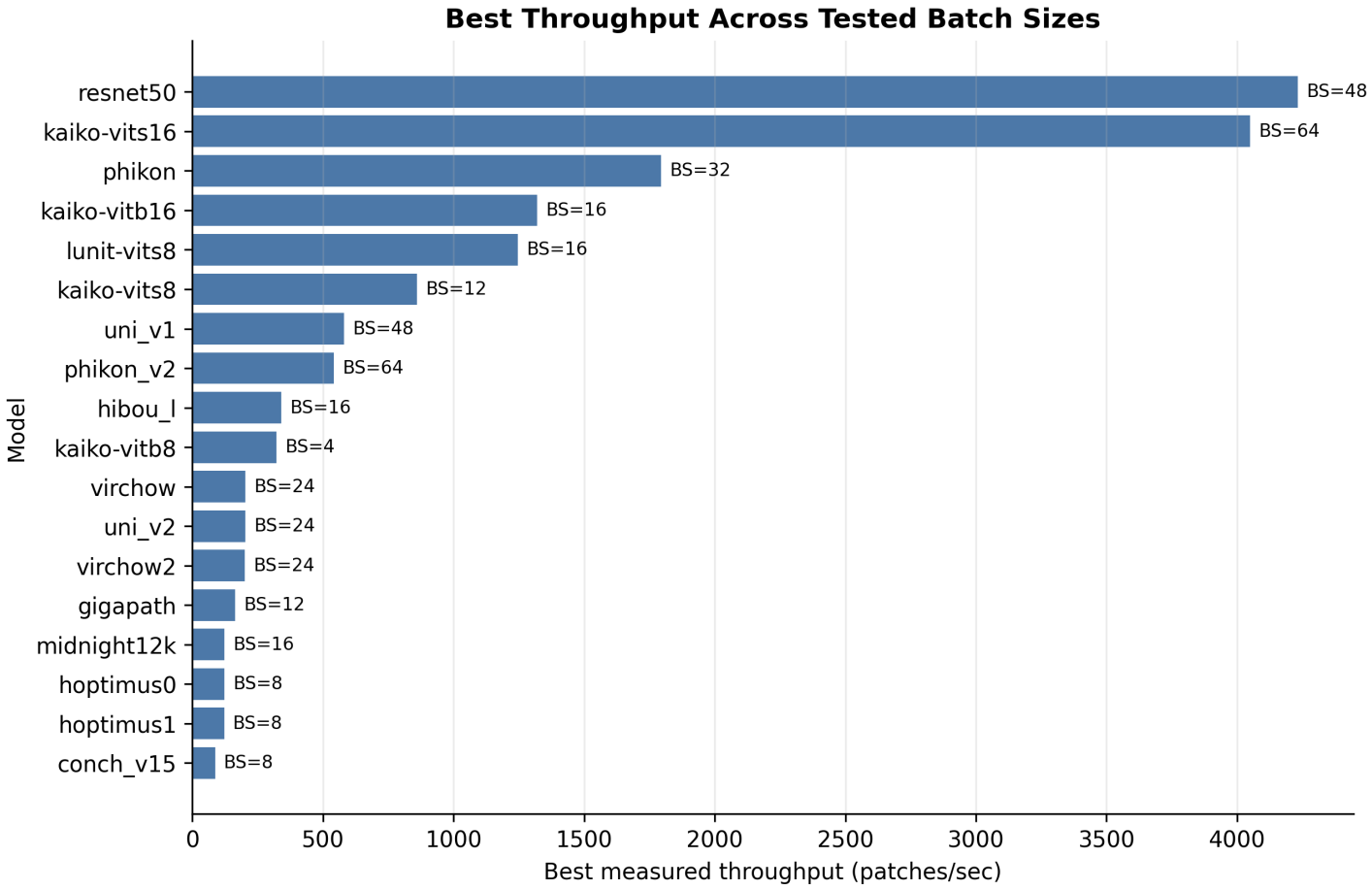


Figure 4: **Vision throughput benchmark.** Best observed throughput for each vision encoder from the local runtime reports. This panel supports the compute-value analysis and helps interpret encoder-specific extraction cost.

Table 7: **Patch-level Encoder benchmarking and avoided inference burden.** Benchmark GPU-hour estimates are computed from final patch counts and measured throughput. Observed logged extraction time from the runtime summary is reported separately in the text.

Encoder	Patches encoded	GFLOPs/patch	Throughput patches/s	Est. FLOPs	Benchmark GPU hours
UNI	334,297,449	59.7	580.9	20.0 EFLOPs	159.9
UNI2-h	334,297,449	180.4	202.7	60.3 EFLOPs	458.1
CONCHv1.5	68,260,887	312.2	87.0	21.31 EFLOPs	217.95
Virchow2	334,297,449	164.6	200.9	55.0 EFLOPs	462.2
Prov-GigaPath	334,297,449	223.4	163.9	74.7 EFLOPs	566.6
H-Optimus-1	334,297,449	296.0	121.8	99.0 EFLOPs	762.4
Phikon-v2	334,297,449	59.7	542.2	20.0 EFLOPs	171.3
ResNet50	260,050,307	4.3	4,233.2	1.1 EFLOPs	17.1

Table 8: **Supported downstream sanity-check/validation performance.** Image experiments used patient-level training, validation, and testing data partitions. Text experiments used stratified group cross-validation with TCGA case identifiers as groups. Values are baseline sanity checks for the released embeddings and should not be interpreted as definitive benchmarks.

Task	Feature set	Modality	Accuracy	Macro-F1	Macro-AUROC	Notes
TCGA pan-cancer	UNI-2h_{20x}	Image	0.813	0.785	0.892	43-class, 70/10/20 partitions
TCGA pan-cancer	Qwen-clean	Text	0.941	0.924	0.9876	8,383 reports; 5-fold CV
CPTAC pan-cancer	UNI-2h_{20x}	Image	0.884	0.854	0.9452	9-class; 70/10/20 partitions

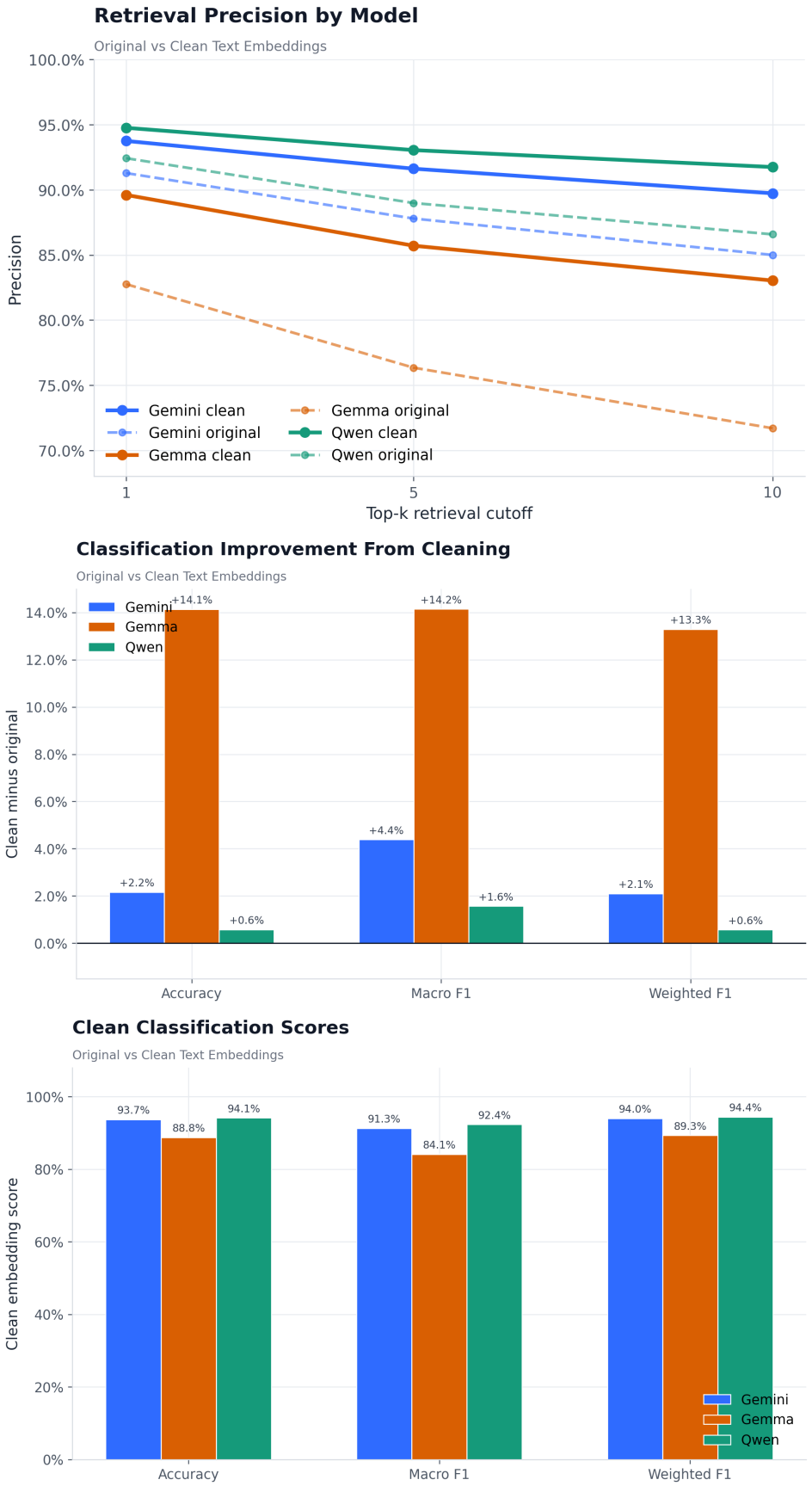


Figure 5: Text embedding validation. Top: nearest-neighbor retrieval precision by embedding model. Middle: classification improvement after diagnostic-report cleaning. Bottom: clean-report classification performance across evaluated Text embedding models.