Retrieval Augmented Generation for Pathology Reports

Aayush Shivashankar, Tanya Nair, Valmik Nahata, Connor Friedman, Zarif Azher, Joshua J. Levy
Emerging Diagnostic and Investigative Technologies, Department of Pathology, Dartmouth Hitchcock Medical Center

ABSTRACT

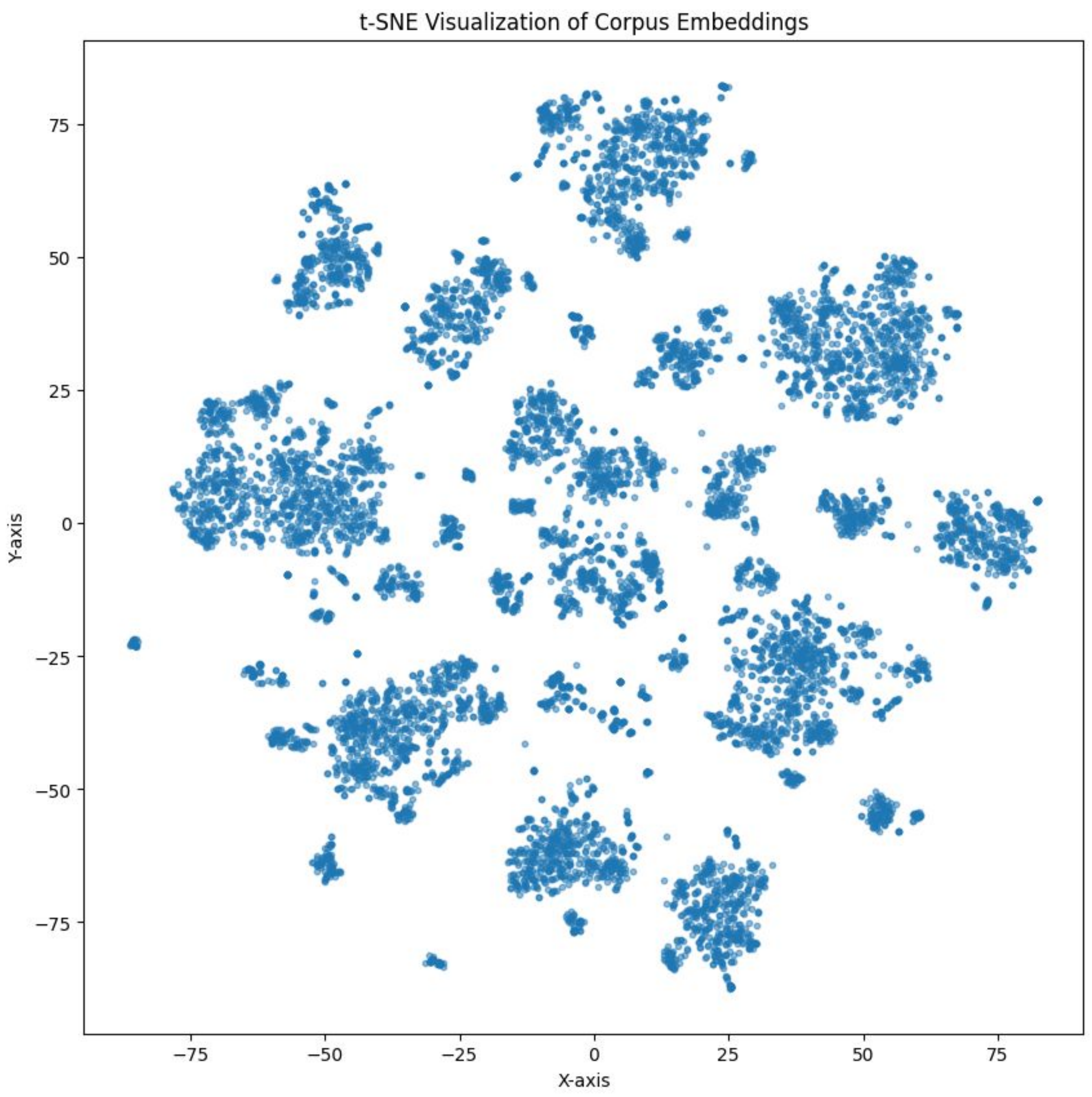
- Clinicians are expected to cross-reference pathology reports to gain a better understanding of a patient's report, especially with different cancer variants
- Pathology reports are not standardized and are written with complexity, making them difficult to understand
- Retrieval Augmented Generation (RAG) model on a Small Language Model (SLM) and fusion search to efficiently retrieve information from pathology reports

INTRODUCTION

- **Clinicians' Barriers to Understanding**
  - Pathologists manually inputting results in pathology reports may leave room for excessive interpretability due to complex wording and phrases of uncertainty
  - Clinicians aiming to cross-reference reports may find it difficult when left with varying understandings of each report
  - Patients cannot receive adequate details on their pathology report unless a clinician has an accurate and holistic understanding of a patient's report
  - Standardized information extraction is necessary for an increase in medical comprehension by clinicians
- **Existing Solutions**
  - Named Entity Recognition systems built from Large Language Models (LLMs)
  - Deep Learning classification systems that help in extracting and coding information from pathology reports
  - Technologies are currently used to assist patients with interpreting pathology reports
- **Extending Research**
  - Lack of Retrieval Augmented Generation (RAG) with Small Language Models (SLMs)
  - Creating a more efficient and accurate method for information extraction and classification of pathology reports compared to traditional LLM-based approaches

METHODS

- **Goal:** Create a RAG model based on a SLM and fusion search that enables enhanced interpretability of pathology reports through drawing key insights and similarities from relevant reports for the clinician user
- **Data:** 9,523 pathology reports from The Cancer Genome Atlas (TCGA)
- **Algorithmic Methods:**
  - Indexed TCGA pathology reports directly into the corpus for retrieval using HuggingFace transformers Python library
  - Used all-MiniLM-L6-v2 to generate document embeddings and Auto Tokenizer for tokenization of both documents and queries
  - Fusion search method: BM25 and FAISS quantized vector search
  - Used rank-bm25 Python library to conduct semantic similarity search based on query - retrieved top 5 similar documents
  - Retrieved top 10 similar documents using FAISS quantized vector search using the faiss-gpu Python library
  - Top 15 documents fed into SLM – returned insights relevant to the query



The image on the left demonstrates the large variance in the way pathology reports are written. They can be far from each other in terms of content and standardization, making accurate retrieval of information nearly impossible. Different pathologists will have different methods of writing reports, which further complicates the issue.

Figure 1: Visualization of corpus documents' similarity and distribution

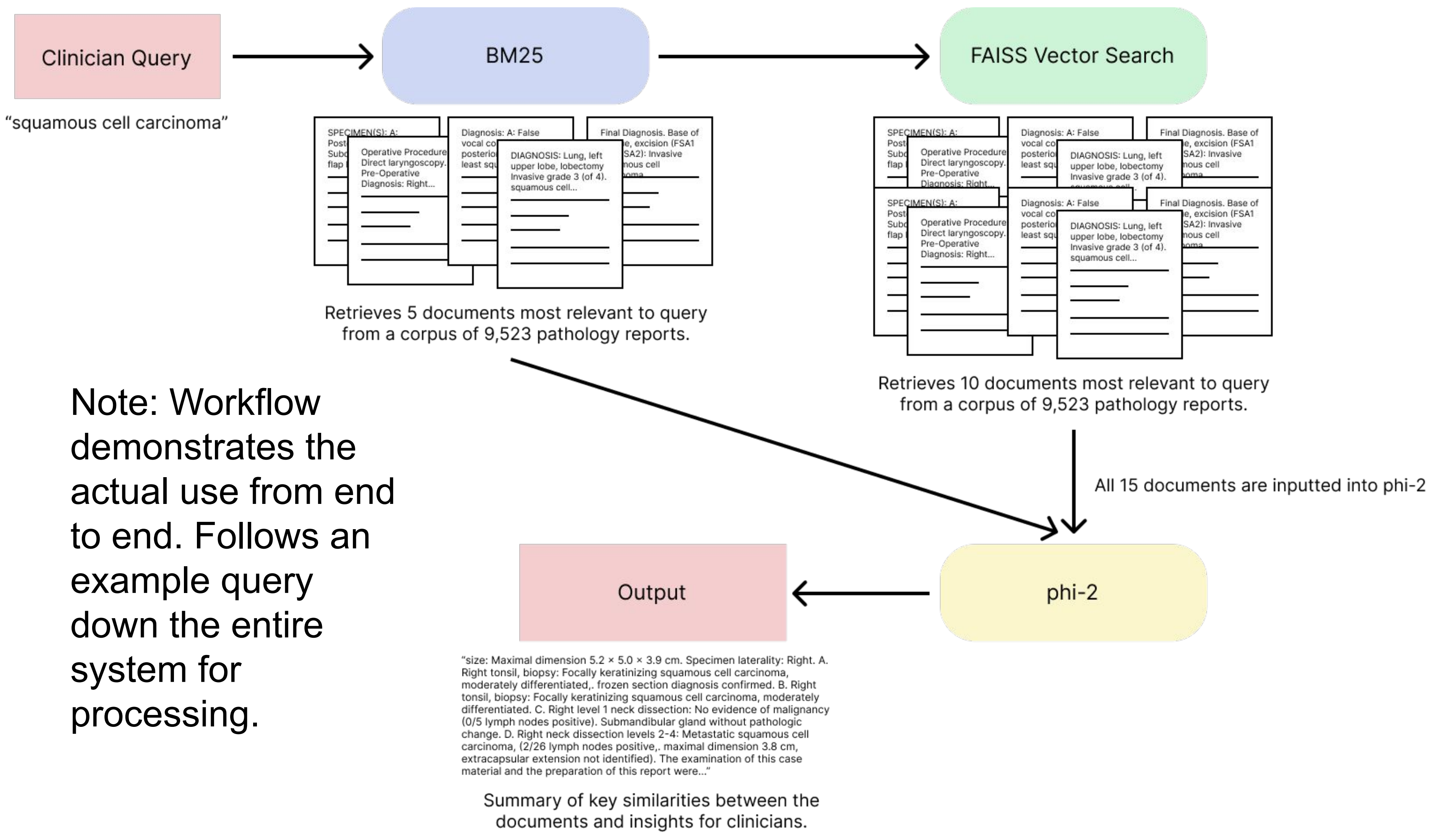
RESULTS

The full corpus with embeddings is 334.5619 GB.

RAM Used By System:	**Naive System RAM:**
6.9057 gigabytes	~100+ gigabytes

Table 1: Memory benchmark data

When uploading the TCGA dataset, there is a large amount of information being processed, therefore the vectors for FAISS search must be quantized. Even with doing this, the corpus totals to a staggering storage. Systems to retrieve large amounts of data can be very time and resource intensive. With the system we developed, the total RAM usage is only 6.9 GB, making it suitable for small devices in a clinical setting without huge hardware modifications to existing tools. A more naive implementation using GPT-4 or other LLM on a local machine could result in over 100 GB usage.


Clinician Query
"squamous cell carcinoma"
BM25
SPECIMEN(S): A:
Operative Procedure Direct laryngoscopy. Pre-Operative Diagnosis: Right...
Diagnosis: A: False vocal co posterio least squ
DIAGNOSIS: Lung, left upper lobe, lobectomy Invasive grade 3 (of 4). squamous cell...
Final Diagnosis. Base of e, excision (FSA1 SA2): Invasive nous cell
Retrieves 5 documents most relevant to query from a corpus of 9,523 pathology reports.
FAISS Vector Search
Retrieves 10 documents most relevant to query from a corpus of 9,523 pathology reports.
All 15 documents are inputted into phi-2
phi-2
Output
Summary of key similarities between the documents and insights for clinicians.

"size: Maximal dimension 5.2 × 5.0 × 3.9 cm. Specimen laterality: Right. A. Right tonsil, biopsy: Focally keratinizing squamous cell carcinoma, moderately differentiated,. frozen section diagnosis confirmed. B. Right tonsil, biopsy: Focally keratinizing squamous cell carcinoma, moderately differentiated. C. Right level 1 neck dissection: No evidence of malignancy (0/5 lymph nodes positive). Submandibular gland without pathologic change. D. Right neck dissection levels 2-4: Metastatic squamous cell carcinoma, (2/26 lymph nodes positive,. maximal dimension 3.8 cm, extracapsular extension not identified). The examination of this case material and the preparation of this report were..."

Note: Workflow demonstrates the actual use from end to end. Follows an example query down the entire system for processing.

Figure 2: Sample model workflow with query "squamous cell carcinoma"

Sample Input: "squamous cell carcinoma"
Sample Output: "size: Maximal dimension 5.2 x 5.0 x 3.9 cm. Specimen laterality: Right. A. Right tonsil, biopsy: Focally keratinizing squamous cell carcinoma, moderately differentiated,. frozen section diagnosis confirmed. B. Right tonsil, biopsy: Focally keratinizing squamous cell carcinoma, moderately differentiated..."

CONCLUSION

Through our methods we were able to demonstrate that it is possible to create a low-powered and low-resource system for the indexing and retrieval of pathology reports. Clinicians will be able to use this system to find pathology reports not only by keyword search, but also through semantic vector search. This means that even if pathology report writing is not standardized between different clinicians, searching will still consider both methods of search: keyword and meaning.

With the specific RAM requirements and benchmarks generated, the system will be capable of running on a Raspberry Pi 5 or similar small computer with an external memory disk. This means that in terms of cost implementation, this system implementation could be easily integrated into hospital systems serving as an out of the box tool that comes with built in security features by simply not needing connection to a network. Because of this, the system design specifically works towards ensuring that even cyber attacks, compromising patient specific data, are not a threat.

Figure 3: Raspberry Pi 5 Visual