KU LEUVEN

FACULTY OF ENGINEERING SCIENCE

Laying the Foundation for Multimodal RAG in Dental Imaging: Evaluating GPT-4o Vision on Oral Panoramic Radiographs

Zhongbo Yao

Thesis submitted for the degree of Master of Science in Artificial Intelligence, specialisation Big Data Analytics

Supervisors

Prof. dr. Peter Claes
Prof. dr. Reinhilde Jacobs

Assessors

Prof. dr. Hilde Bosmans
Dr. Tom Eelbode

Assistant-supervisor

Soroush Baseri Saadi

Academic year 2024 – 2025

Published by Zhongbo Yao,
Department of Computer Science, Celestijnenlaan 200A bus 2402, B-3001 Leuven

Contents

Abstract

This thesis evaluates GPT-4o Vision's ability to interpret oral panoramic radiographs by prompting it to answer thirteen clinical dental questions and propose the most likely diagnosis. The thesis systematically explores a range of prompt-engineering techniques, including few-shot, chain-of-thought, tree-of-thought, self-consistency, self-debate, and self-critique, to determine whether they enhance GPT-4o Vision's performance in the dental domain. Next, two practical usage scenarios are simulated to assess the model's reliability in real-world conditions. Across these experiments and accompanying ablation studies, it is found that GPT-4o Vision relies heavily on textual priors rather than genuine visual information, resulting in suboptimal performance on dental image interpretation. Given the limited dataset (25 cases) and budget constraints, the results indicate the need for further domain-specific fine-tuning and additional evaluation. Future work should include larger case collections, expert clinician annotations to clarify inter-question dependencies, and a deeper analysis of GPT-4o Vision's error patterns by professional doctors to guide improvements.

List of Figures and Tables

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

1.1.1 Transformers, LLMs, and Healthcare

Transformer architectures and their large-scale next-token training have driven a recent surge in language model capabilities. Models such as GPT-4, Claude, and Grok (large language models, LLMs) leverage self-attention to capture long-range dependencies across billions of parameters. Their conversational interfaces (e.g., ChatGPT) can generate coherent text, answer complex questions, and even produce code, all with minimal effort required for prompt engineering.

Large language models are integrated into clinical workflows to streamline routine tasks and enhance provider efficiency. For example, they can generate concise patient summaries from electronic health records, draft and revise discharge or referral letters, extract and standardize key findings from clinical notes, analyze patient-reported symptoms through conversational interfaces, and deliver on-demand answers to physicians' diagnostic and treatment queries [1]. By automating these documentation and information-retrieval tasks, LLM-powered tools help free up clinicians' time for direct patient care.

1.1.2 Clinical Importance of Panoramic Imaging

In dentistry, panoramic radiographs (orthopantomograms) serve as the standard first-line imaging modality, comprehensively capturing the entire maxillofacial region, including jaws, teeth, and surrounding bones. These images are crucial for detecting and characterizing oral pathologies, including cysts, tumors, bone lesions, impacted or supernumerary teeth, and anatomical anomalies. Clinicians routinely assess key diagnostic features, such as lesion count, maximum size, border definition, and spatial relationships to adjacent teeth and cortical bone, to inform treatment planning.

Integrating LLMs into this workflow could automate the extraction and annotation of these features, producing structured reports in real-time. Beyond raw measurements, an LLM-powered assistant can generate data-driven insights and differential diagnoses, offer a real-time second opinion to reduce diagnostic oversights, and function as an interactive educational tool for dental trainees, reinforcing pattern recognition and clinical reasoning during image review.

1.1.3 Limitations of LLMs

General-purpose LLMs excel at broad tasks, such as email completion, article summarization, and everyday object identification. However, their knowledge is confined to the text and images encountered during pretraining. They also lack the structured, professional reasoning processes, such as differential diagnosis or evidence-based inference, that clinicians employ, as LLMs generate text by predicting the most statistically likely next word without explicit hypothesis generation or reasoning [2]. As a result, when these models encounter clinical or radiological queries (for example, identifying oral pathologies on panoramic images), they often produce incomplete or superficial answers. This shortfall stems from a mismatch between the models' generalist pretraining objectives and the deep, domain-specific reasoning required in healthcare, compounded by a scarcity of medical imaging data.

Empirical benchmarks illustrate this weakness. In one study on a Taiwanese emergency medicine exam, GPT-3.5 and GPT-4 scored only around 40–50 % without further adaptation, whereas fine-tuned or retrieval-augmented variants exceeded 60 % [3]. Similarly, a dedicated clinical scribing system (DeepScribe) outperformed GPT-4 [4], indicating the need for targeted fine-tuning or retrieval strategies to achieve reliable performance in high-stakes medical settings.

To address this gap, retrieval-augmented generation (RAG) has emerged as a promising solution. RAG combines a retrieval module that queries an external knowledge base or document store for relevant, up-to-date information with the LLM's generative capabilities, allowing the model to reason over its pretrained knowledge and the freshly retrieved content.

1.1.4 Text-Only Retrieval-Augmented Generation

Retrieval-augmented generation (RAG) enhances an LLM's static, pretrained knowledge by pairing it with an external document store. At inference, a retriever first identifies the most relevant passages from a domain-specific corpus; the LLM then conditions its generation on both these retrieved texts and its internal parameters, which was first formalized by Lewis et al. [5]. Because the knowledge base can be updated independently of model fine-tuning, RAG enables real-time access to the latest guidelines, case reports, or literature without retraining the underlying language model. In medicine, text-only RAG has proven its value by improving the accuracy of question answering [6], enhancing information extraction and chart

summarization from electronic health records [7], and automating radiology report drafting [8].

However, many clinical tasks rely on non-textual data, such as images, audio recordings, and structured tables, which carry richer information than pure texts. Standard textual RAG pipelines only retrieve text and cannot directly process these modalities. Multimodal RAG encompasses strategies for integrating any combination of text, images, audio, and structured data to support richer retrieval and generation. This thesis focuses on the visual branch of multimodal RAG, which combines direct image encoding with text retrieval to build a unified framework that more accurately interprets panoramic dental images in context.

1.1.5 Multimodal RAG: State of the Art

Multimodal RAG enhances traditional text-only retrieval by allowing the model to utilize images in the same manner as it uses documents, either by first captioning the image or by incorporating image embeddings alongside text, enabling the retrieval of clinical references based on images. In a dental setting, it is expected that a multimodal RAG assistant could enable automated, up-to-date annotation of panoramic radiographs, highlighting lesion boundaries, proposing likely diagnoses by retrieving similar case reports, and even recommending next-step imaging or management guidelines. By grounding its image interpretations in real clinical documents, such a system should offer general dentists rapid, evidence-backed second opinions, reduce variability in radiographic reporting, and serve as an on-demand tutor for dental trainees learning to recognize subtle oral pathologies.

1.2 Problem Statement & Challenges

1.2.1 Challenges of Multimodal RAGs on Oral Panoramic Images

In text-only RAG, retrieval quality depends on how well the retriever matches a query to stored documents. In a multimodal setting, an equally critical factor is the vision encoder's ability to produce medically meaningful representations of the input panoramic radiographs. Any errors in interpreting features, such as lesion morphology, cortical changes, or tooth relationships, will directly degrade downstream retrieval and generation.

Most vision encoders (e.g., ViT, CLIP) are trained on generic natural-image datasets and struggle to capture the subtle radiographic patterns in oral panoramic images. In domains such as neuro and thoracic imaging, this gap is routinely closed by fine-tuning on large, domain-specific datasets to align image embeddings with clinical terminology [9]. Unfortunately, oral radiology still suffers from data scarcity for effective fine-tuning, and no pretrained models are available for automatic lesion annotation on panoramic dental radiographs [10].

1.2.2 Thesis Specific Challenges

Fine-tuning a vision encoder generally demands large, domain-specific datasets. But only 25 annotated panoramic cases are available here for this thesis, far too few to train a reliable ViT or classification head (Appendix A). Budget constraints further limit the number of experimental runs. Consequently, this thesis focuses on GPT-4o Vision's zero-shot and few-shot performance: for each of fourteen clinical questions, the model generates structured answers (using various prompt engineering techniques) in a single run, justified by the low variance observed in GPT-4o Vision's outputs (Experiment 4.2.3). Performance is evaluated via weighted recall against doctor-verified ground truth (Section 3.5).

1.3 Research Question

This thesis investigates how accurately GPT-4o Vision can analyze and interpret oral panoramic radiographs of jawbone lesions. Specifically, the thesis evaluates its ability to extract structured radiological features, such as lesion count, size, border characteristics, anatomical relationships, etc. Then, it validates those outputs against clinician-annotated ground truth across a range of prompting strategies (e.g., zero-shot, few-shot, chain-of-thought, tree-of-thought, self-consistency, etc.).

1.4 Contributions

This thesis makes the following key contributions:

- **First GPT-4o Vision Benchmark on Oral Panoramic Images:**
  The thesis presents the first systematic evaluation of GPT-4o Vision on 25 real-patient oral panoramic images, measuring its ability to extract 13 structured radiological features (e.g., lesion count, size, borders, anatomical relationships) and then to derive a diagnosis against clinician-verified ground truth.
- **Prompting Strategy Comparison:**
  The thesis compares zero-shot, few-shot, chain-of-thought, tree-of-thought, self-consistency, self-debate, and self-critique prompting techniques to identify which paradigms maximize GPT-4o Vision's image-to-text reasoning performance.
- **Robustness Analyses:**
  Through leave-one-out and error-correction studies, this thesis examines GPT-4o Vision's resilience to missing or incorrect context. The thesis further performs ablation experiments to assess whether the model can still predict answers from text alone and temperature & question-order sensitivity tests to reveal any effect of model hyperparameters and internal positional dependencies. These analyses collectively demonstrate GPT-4o Vision's heavy reliance on textual findings rather than visual understandings.

- **Practical Insights:**
  The thesis demonstrates that, without domain-specific fine-tuning, GPT-4o Vision alone is not yet reliable for clinical deployment in dental radiology. The thesis concludes with guidelines for integrating LLMs into full multimodal RAG pipelines and outlines future work to understand GPT-4o Vision performance more deeply and to improve it for better integration into multimodal RAG models.

1.5 Thesis Organization

The remainder of this thesis is organized as follows:

- **Chapter 2: Literature Review**
  Reviews foundational work on large language models, retrieval-augmented generation, multimodal methods in medical imaging, and prompt engineering techniques.

- **Chapter 3: Methodology**
  Describes the 25-case panoramic image dataset, prompt engineering strategies, experimental protocols, and evaluation metrics.

- **Chapter 4: Experiments and Results**
  Presents benchmarking of GPT-4o Vision, prompt of engineering effects exploration, temperature and question order tests, real usage scenarios simulation, and ablation study.

- **Chapter 5: Conclusion, Limition and Future Work**
  Interprets the findings, outlines limitations, and proposes further work to understand GPT-4o Vision's limitations more deeply and to establish a fully generative multimodal RAG system in dental radiology.

Chapter 2

Literature Review

2.1 Modern NLP Development

Natural-language processing (NLP) began with statistical models, such as n-gram models [11] and recurrent neural networks like LSTM [12], which can capture short contexts due to their ability to process sequential data. However, they struggled to capture long-distance language dependencies and contexts, suffering from problems such as gradient vanishing or explosion [13]. In 2017, Vaswani et al. came up with a paper named 'Attention is All You Need', introducing transformer architecture [14] revolutionized the field by replacing recurrence with self-attention mechanism, firstly invented by Bahdanau et al. [15], enabling both efficient parallel training and excellent context understanding. Scaling that architecture to billions of parameters gave birth to the first large-language models (LLMs), such as OpenAI's GPT-1 and GPT-2 [16, 17], which demonstrated unprecedented fluency and few-shot learning capabilities. Following these pioneers, models like Grok [18], ChatGPT [19], and Gemini [20] have been invented, with excellent performance in completing various text-based tasks.

2.2 LLMs Bottleneck and RAGs

However, many bottlenecks of LLMs are also observed. Kaplan et al. [21] observed that the model size must increase exponentially to achieve the same gain, measured by cross-entropy. However, with the already huge number of parameters, current LLM models are actually underfitted, as argued by Hoffmann et al. [22]. Guu et al. [23] found that storing all knowledge in models' weights is very inefficient, and then they introduced REALM, which integrates an information retriever into LLMs during pretraining, a type of early retrieval-augmented-generation (RAG) models. Similarly, Lewis et al. and Karpukhin et al. also introduced RAG ideas [5, 24], laying the foundation for the development of later RAG models.

2.2.1 Text Based RAGs

Text-based RAG models have already seen many successes in practical applications, including the medical domain. For example, Sohn et al. [6] introduced RAG^2, which filters retrieved information according to learned rationales and balances across four different medical corpora, achieving an improvement of up to 6.1% on medical question answering benchmarks. Shi et al. [25] proposed a RAG pipeline for injecting knowledge from the medical knowledge base to queries for retrieval augmented generalization. Zhang et al. [26] further improved RAG performance by employing a 'map-reduce' strategy to split the retrieved documents into smaller pieces, thereby mitigating the information loss problem. To increase the RAG accuracy and reduce its hallucinations, Sun et al. [27] employed RadGraph to extract knowledge from medical reports to generate the training dataset, on which their model is trained. The method enables the dataset to align more closely with the facts.

2.2.2 Multimodal RAGs

Following the success of textual LLMs, vision LLMs are expected to revolutionize the computer vision field as well, which was previously dominated by CNN-based models. Two foundational transformers with vision ability proved the feasibility with excellent performance. ViT [28] was proposed by Dosovitskiy et al., which is built upon the text-based foundational transformer BERT [29], bringing the opportunity to understand images using transformers into public sight. ViT tokenizes image patches instead of text and the model is trained on a target classification task. Then, CLIP was proposed by Radford et al. [30], which aligns text descriptions with images, enabling the use of transformers in multimodal tasks.

Encode Images to the Latent Space

Then, there are numerous emerging efforts on multimodal RAGs so that RAGs can understand not only text information but also information in images. The first mainstream approach is to embed images to a latent space shared with the text embeddings, allowing retrieval to be completed across modalities using cosine similarity. The primary objective of this method is to accurately align the image embeddings with textual embeddings in the latent space.

Chen et al. [31] encoded images and texts separately, then concatenate the embeddings for fusion using a T5 encoder. Likelihood is maximized to minimize the misalignment. Unlike fine-tuning an encoder for fusion after text and image embeddings have been generated, Zhou et al. [32] focused on training image tokenizers before the embedding step. The idea is to ask tokenizers to be responsible for multimodality alignment instead of fusion encoders. The method saves great resources for fine-tuning the model. Zhang et al. [33] proposed using unsupervised contrastive learning to train multimodal encoders, in which texts naturally appear with images. This strategy saves great manual effort and performs well in embedding images. Liu et al. [34] introduced the insertion of gated cross-attention-dense blocks into vision

encoders during the fine-tuning of multimodal RAG models. During the fine-tuning phase, only the newly inserted blocks are updated. The method shows a promising result.

Wei et al. [35] proposed a new multimodal retriever, which is trained on the M-BEIR dataset constructed by the authors. The combination of CLIP or BLIP with three different fusion methods of vision embeddings is explored. The results show that using CLIP combined with linear aggregation of image and text embeddings yields the best and most consistent results. Zhang et al. [36] also trained a retriever based on a dataset proposed by themselves. The datasets highlight the relationship between two images generated by LLMs, depicting relationships among images from the same website page. The retriever combines the text embeddings with image embeddings using self-attention blocks, followed by an attention-pooling layer. Yang et al. [37] employed a multimodal retrieval LLM to generate captions for images by asking the query to pay attention to both the query image embedding and the retrieved caption embedding using gated cross-attention blocks.

Summarize Images using Text

Another approach is to summarize images using text descriptions and then perform textual retrieval in the RAG pipelines. The primary focus of this approach is to generate text descriptions based on images accurately. It can be achieved by the fine-tuning or careful prompt of engineering. And they are usually used together in this scenario.

Anand et al. [38] rewrote original queries using LLMs by providing the original query and relevant contexts retrieved from documents. Then, the rewritten queries are used for fine-tuning rerankers, which largely improves the reranker's performance. Thus, the generation performance is improved due to the retrieval performance increase. Nogueira et al. [39] trained a sequence-to-sequence model to predict possible queries of a document. Then, the document is enriched by the predicted queries. However, Gospodinov et al. [40] found that some of the predicted queries are not relevant to the document, thus leading to hallucinations. Based on Nogueira's work [39], they further filter the predicted queries by only keeping a proportional of the queries that are the most similar to the documents, improving the retrieval effectiveness and reducing the indexing time. Chan et al. [41] directly trained a model to generate refined queries and to generate good responses according to the refined queries. This supervised-like approach relies on the labeled refined queries generated by ChatGPT, which can be the bottleneck.

2.3 Prompt Engineering

There are also efforts in prompt engineering without training or fine-tuning models. Cho et al. [42] treated pages of PDF files as images and embed them directly. They expect the generated multimodal embeddings, which contain both image and text

information of the pdf page, to preserve more information so that RAG can achieve a better result. Koo et al.[43] enhanced original queries using LLMs based on few-shot tasks. The prompt for rephrasing contains the original query, retrieved relevant documents related to the original query, and examples of rephrased queries, along with their scores measured by a hybrid approach combining BM25 and dense similarity. The paper presents an interesting idea of providing LLMs with examples to improve their rephrasing performance. Agbareia et al. [44] explored how various prompt engineering techniques affect multimodal LLM performance on glaucoma diagnosis, measured by sensitivity, NPV, etc. In the paper, various prompt engineering techniques are applied to different multimodal large language models (LLMs), and a conclusion is reached that a significant performance improvement is observed when applying prompt engineering with example images.

2.4 GPT Vision Models on Medical Applications

Existing vision models are also explored to see if they can generate textual descriptions about medical information directly without fine-tuning and training. Zhan et al. [45] explored the possibility of utilizing GPT-4o to generate structured medical reports given images and the corresponding description in natural language. An auto-prompt engineering pipeline is used, where the first prompt generated by GPT is optimized by doctors and then used to generate structured reports. The paper demonstrates that ChatGPT is capable of generating structured reports, albeit with some flaws, and that human efforts are necessary for the prompt engineering stage. Zhang et al. [46] and Brin et al. [47] assessed the performance of using GPT-4o to generate several answers to specific questions according to the provided radiological image, and a high accuracy of answers is observed. However, the experiment only includes three easy questions, leading to a relatively good performance.

2.5 Model Selection

Given the limited size of the dataset, this thesis evaluates whether existing multimodal GPT models can accurately interpret dental panoramic images without additional training or fine-tuning. Several prominent models—Claude, LLaVA, Qwen-VL, ChatGPT, Grok, DeepSeek, and Gemini—were considered. Rather than performing an exhaustive comparison to establish a leaderboard, the goal is to select a single model known for robust zero-shot performance. Furthermore, an experiment is performed in Experiment 4.1 proving GPT-4o Vision has the best performance compared with Grok and Claude. For these reasons, GPT-4o Vision is chosen for annotating dental panoramic images: it offers an easy access to API, excellent zero-shot accuracy, extensive community support, and comparatively low per-token costs.

2.6 Research Gap

Despite advances in multimodal LLMs for general medical imaging and radiology, zero-shot interpretation of dental panoramic images remains largely unexplored. Most prior studies have focused on broad anatomical images or constrained tasks with limited question sets, often relying on fine-tuning or retrieval-augmented pipelines to achieve acceptable accuracy. Dental panoramic radiographs, however, present unique challenges, such as limited dataset and subtle differences among different lesions. As a result, it is unclear whether a current multimodal model can reliably identify clinically relevant features in panoramic dental views without any domain-specific training. By systematically evaluating GPT-4o Vision's zero-shot performance on a curated set of dental panoramic cases, this thesis fills that gap: it measures weighted score against expert annotations, analyzes common failure modes, and assesses whether GPT-4o Vision's existing vision capabilities can generalize to odontological radiography without fine-tuning.

Model	Dev.	Vision Backbone	Accessibility	Vision Ability	Image Token Cost (Input+Ouput)
Claude Sonnet 4	Anthropic	ViT	API (invite-only)	Yes	$18 / 1M tokens
LLaVA	UCSD / MSR	ViT-GPT fusion	Open-source (GitHub)	Yes	Free
Qwen2.5-VL	Alibaba	Qwen Vision	API (China-centric)	Yes	Free
GPT-4o Vision	OpenAI	CLIP	API & Chat UI	Yes	$12.5 1M tokens
Grok	xAI	ViT	API	Yes	$18 / 1M tokens
DeepSeek-V3	DeepSeek-AI	ResNet + ViT	API (High latency)	No	-
Gemini 2.5 Pro	Google	ViT	API	Yes	$11.25 / 1M tokens

TABLE 2.1: Features across different famous GPT models comparison.

[48, 49, 50, 51, 52, 53]

Chapter 3

Methodology

This chapter provides a detailed overview of the underlying background, describes the dataset and experimental setup in full, and defines the evaluation metrics used to assess model performance.

3.1 Theoretical Foundations

3.1.1 Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) enhances a language model's static, pre-trained knowledge by coupling it with an external document store. At inference, a dense retriever first selects the top-k passages most relevant to the user's query that is measured via vector similarity. Then, these retrieved snippets are prepended to the prompt before the LLM generates its answer. By fetching up-to-date or domain-specific information on the fly, RAG overcomes the limitations of memorized parameters. It enables real-time access to new guidelines, case reports, or research articles without requiring retraining of the underlying model.

A typical RAG pipeline comprises the following major stages:

- **Document Chunking:**
  Split raw documents into smaller passages to improve retrieval granularity using fixed-length, semantic, or sliding-window splits.
- **Embedding:**
  Encode each passage and the query into vectors, using dense models (e.g., dense retriever like DPR[24]) for semantics or sparse methods (e.g., BM25[54]) for keyword matching.
- **Retrieval:**
  Compute similarity (typically cosine) between the query vector and each passage embedding, then select the top-k contenders.

- **Reranking (Optional):**
  Apply a cross-encoder or neural ranker over query-passage pairs for more precise relevance scoring.
- **Augmentation & Generation:**
  Concatenate the chosen passages with the original query and feed this into the LLM, which generates a response grounded in the retrieved context.

Fig 3.1 illustrates the basic structure of textual RAG models [55].

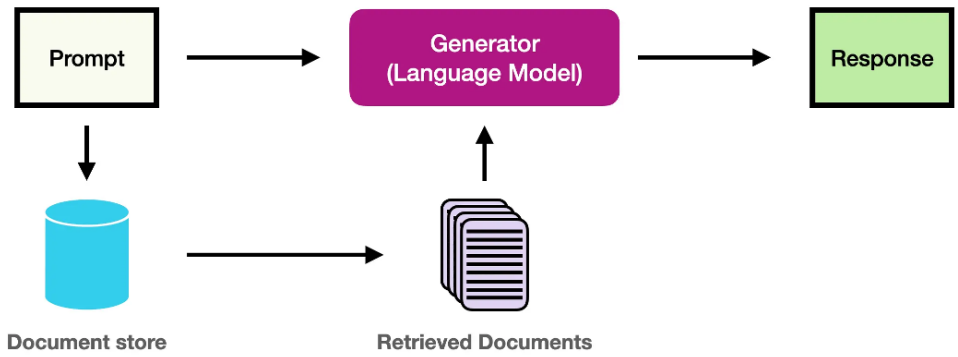

FIGURE 3.1: RAG structure.

Extending RAG to multimodal inputs entails two principal strategies. In the caption-based approach, a vision-to-language model (e.g., BLIP[56]) generates a natural language summary of the image, and standard text retrieval is then performed on that caption. In the embedding-based approach, a vision encoder (e.g., CLIP's ViT[28]) maps the image into the same high-dimensional space as document embeddings, enabling direct cosine similarity retrieval between image and text vectors. Captioning may omit fine-grained visual details, whereas embedding-based retrieval depends critically on the encoder's ability to capture domain-specific features.

3.1.2 GPT-4o Vision: CLIP + LLM Backbone

GPT-4o Vision is the multimodal extension of GPT-4 that combines a CLIP-style vision encoder with the GPT-4 decoder, allowing it to ingest images and text jointly and generate coherent, context-aware responses [30]. By prepending the CLIP-derived image embedding as a vision token, GPT-4o seamlessly attends to visual and textual inputs within its autoregressive transformer framework.

CLIP

CLIP (Contrastive Language–Image Pretraining) learns a shared embedding space for images and text by jointly optimizing a text encoder (e.g., based on BERT [29])

and an image encoder (e.g., based on ViT [28]). Given a batch of N image–text pairs $\{(I_i, T_i)\}_{i=1}^{N}$, each image I_i is passed through ViT to produce a normalized visual embedding v_i, and each text T_i is passed through the BERT-style encoder to produce a normalized textual embedding t_i. The model then computes similarities $s_{ij} = \langle v_i, t_j \rangle$ and minimizes the symmetric contrastive loss

$$\mathcal{L}_{\text{CLIP}} = -\frac{1}{2N} \sum_{i=1}^{N} \Big[\log \frac{\exp(s_{ii}/\tau)}{\sum_{j=1}^{N} \exp(s_{ij}/\tau)} + \log \frac{\exp(s_{ii}/\tau)}{\sum_{j=1}^{N} \exp(s_{ji}/\tau)} \Big],$$

where τ is a learnable temperature, pulling matched pairs together and pushing non-matches apart.

CLIP retains a special [CLS] token embedded in each encoder that summarizes the entire input. Particularly, v^{CLS} represents images and t^{CLS} represents texts. In GPT-4o Vision, the image CLS vector is prepended as a vision token to the decoder's input sequence, allowing the transformer to jointly attend to visual and textual information without further fine-tuning the vision encoder.

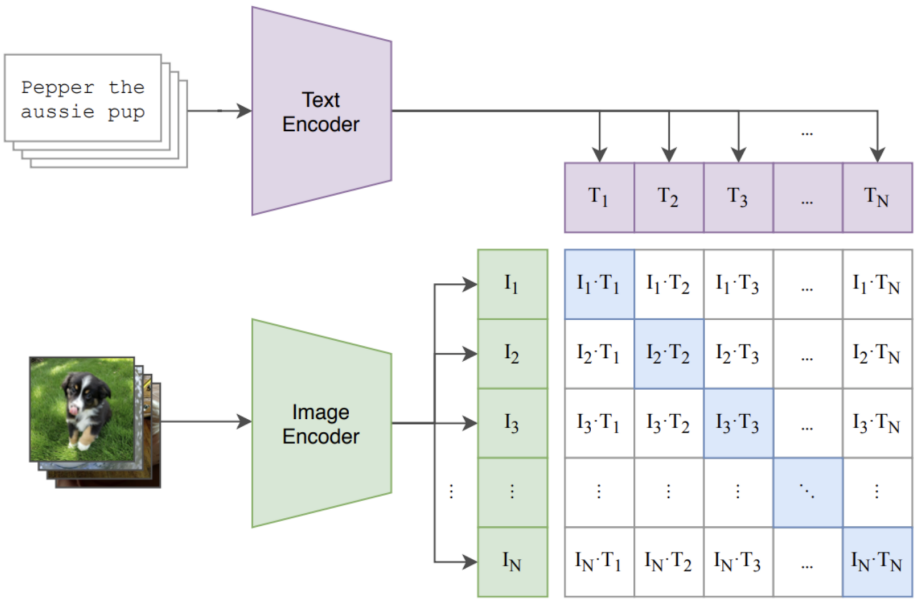

Figure 3.2: CLIP dual-encoder contrastive pretraining. Image and text inputs are encoded to CLS vectors, whose pairwise similarities are optimized via a symmetric contrastive loss.

GPT-4 Backbone

GPT-4 [57] is a decoder-only Transformer that builds directly on the attention architecture[14] and popularized by GPT-3's few-shot learning demonstrations[2].

Trained autoregressively on a diverse mixture of web text, code, and other high-quality corpora, GPT-4 scales up parameter count, depth, and training data to achieve state-of-the-art performance on a wide range of language tasks. Its core mechanism remains next-token prediction: given a sequence of tokens (now including a prepended vision CLS token in GPT-4o Vision), the model applies masked self-attention and feed-forward layers to compute the probability distribution over the next token. Optimizations in training, architecture, and inference further improve latency and cost, making GPT-4 the foundation for multimodal systems that seamlessly attend over both visual and textual inputs.

3.1.3 Prompt Engineering

The power of prompt engineering is explored in the thesis to steer GPT-4o Vision toward more accurate, transparent, and reliable interpretations of panoramic radiographs. By carefully designing input prompts, it can (1) explicitly ground the model's answers in the visual evidence, (2) expose its internal reasoning process for greater interpretability, and (3) filter out spurious associations or "hallucinations" that commonly afflict LLMs [58, 59].

In practice, prompt engineering works by supplying GPT-4o Vision with structured information, such as question–answer pairs, step-by-step reasoning examples, or multiple independent reasoning attempts, that clarify the desired output format and logical flow [60]. These information or examples reduce ambiguity, align the model's latent attention patterns with domain-specific features (for example, lesion margins or tooth relationships), and help the model self-correct when it deviates from the intended path [61]. Rather than discovering new knowledge in its weights, GPT-4o Vision uses these carefully crafted prompts to simulate professional reasoning, yielding more consistent and coherent generations.

The prompt engineering techniques used in this thesis are explained as following:

1. **Zero-shot Prompting**
   Zero-shot prompting asks GPT-4o Vision to generate diagnostic annotations from a single instruction with no in-context examples or reasoning cues. This method is often used to assess a model's raw, pretrained capabilities on novel tasks [2]. In a clinical imaging setting, zero-shot performance indicates whether the model has internalized enough radiological knowledge when fine-tuning data are scarce. It also minimizes engineering effort and API costs, making it practical for resource-constrained environments.

   However, zero-shot prompting has clear limitations. Without examples, the model may misinterpret output format or necessary detail, leading to inconsistent responses. In specialized domains like oral radiology, it cannot convey nuanced terminology or conventions, so accuracy is typically lower than even simple few-shot baselines. The lack of grounding examples also increases hallucination and reasoning drift since there is no pattern to follow. Finally,

zero-shot learning lacks calibration context, causing the model to over- or under-commit to uncertain inferences. Thus, while useful as a baseline, zero-shot prompting is insufficient for complex, domain-specific tasks.

2. **Few-Shot Prompting:**
   Few-shot prompting augments the base template with several (question, answer) pairs that demonstrate the desired reasoning, output format, and domain-specific terminology. In panoramic lesion analysis, examples include correctly formatted JSON for lesion count, size, and other features. These in-context demonstrations guide the model toward the required granularity and structure, reducing off-topic responses and hallucinations [2]. Empirically, few-shot prompts will always boost GPT-4o Vision's performance over zero-shot baselines.

3. **Chain-of-Thought with Few-shot Prompting:**
   Chain-of-Thought (CoT) prompting asks the model to explain its intermediate reasoning steps before giving a final answer, similar to how clinicians think through cases step by step. In zero-shot CoT, users simply add "Let's think step by step" to the base prompt, encouraging GPT-4o Vision to break down its visual and clinical judgments in sequence [58]. Few-shot CoT goes further by showing one or two worked examples, such as how to spot cortical expansion, count lesions, and consider differential diagnoses, so the model learns not just the format but the depth of analysis needed [62].

   CoT can significantly improve performance on tasks requiring multiple inferences by turning a complex problem into smaller steps and preventing the model from jumping to incorrect conclusions. However, CoT has drawbacks in that the reasoning steps may still reflect the model's inherent biases, which sometimes makes the generation worse.

4. **Tree-of-Thought Prompting:**
   Tree-of-thoughts prompting extends the chain-of-thought by treating internal reasoning as a search over a branching tree of partial "thought" states rather than a single linear sequence. At each step, the model generates multiple candidate continuations, which can be mini-explanations or inferences, and then scores each one accordingly. Only the top-scoring branches are retained for further expansion, and any branch that dead-ends or violates domain constraints can be pruned.

5. **Self-Consistency:**
   Self-consistency aggregates multiple independent reasoning traces to improve answer reliability. Instead of generating a single chain-of-thought, the model is prompted to produce multiple separate CoT outputs for each question. Each run may explore different reasoning paths due to inherent non-determinism, and the final answer is determined by a simple majority vote across all the outputs. This approach smooths over occasional reasoning errors or hallucinations by relying on the most stable consensus answer.

6. **Self-Debate:**
   Self-debate prompts GPT-4o Vision to simulate multiple independent experts, each providing a complete reasoning trace before converging on a single consensus answer. In practice, the model is instructed to role-play, for example, three different dental radiologists reviewing the same panoramic image. Each "doctor" first generates a full chain-of-thought explanation and then states a final label for the given question. After all three experts have provided their individual answers, GPT-4o Vision is prompted to compare these viewpoints, identify any areas of agreement or disagreement, and produce one adjudicated response. This approach is designed to simulate how a panel of specialists might debate a complex case: if one reasoning path goes wrong, another may still be valid, and the final consensus is expected to overwrite individual errors. By aggregating multiple hypothetical expert opinions in this way, self-debate seeks to amplify correct reasoning paths and mitigate the impact of single-chain hallucinations or misinterpretations.

7. **Self-Critique:**
   Self-critique asks GPT-4o Vision to review and refine its own single reasoning chain rather than generating multiple parallel viewpoints. In this workflow, the model first generates a chain-of-thought explanation and a preliminary answer to a question. Immediately afterward, it is prompted to critique its own previous reasoning by identifying any mistakes or missing details. The model then examines its own steps, points out potential oversights (for instance, "I realize I may have overlooked cortical expansion when assessing the lesion border"), and issues a revised final answer that incorporates those corrections.

   Unlike self-debate, which explicitly creates three independent chains and merges them, self-critique operates entirely on the initial single chain. The model attempts to detect and amend its own errors. By relying on a single reasoning path rather than multiple parallel chains, self-critique can be faster than self-debate but may also fail to uncover errors that an entirely separate viewpoint could have identified. Whereas self-debate crowdsources "expert" disagreement to arrive at a consensus, self-critique depends solely on the model's own ability to recognize and correct its mistakes.

3.2 Dataset Description

The dataset comprises 25 real-patient panoramic radiographs (orthopantomograms, PNG format, around 1920 × 1002 px) collected at UZ Leuven, each paired with structured annotations by a professional oral radiologist. An accompanying .xlsx file provides, for each case, fourteen diagnostic questions and the expert's selected answers, along with the patient's demographic information (including race, gender, etc.). However, the patient metadata is not used in this thesis since its aim is to evaluate GPT-4o Vision's ability to answer accurately using image data alone. For the first thirteen questions, the radiologist chose exactly one response from

a predefined multiple-choice list, and the fourteenth question ("What is the most possible lesion?") was answered free-form. The questions and allowed answers are shown in Table 3.2.

Figure 3.3 shows the distribution of the labels across all 25 cases. Notice the strong class imbalance (e.g., "Central" origin in 100 % of cases in Q6) and varying diversity in other questions (e.g., lesion size, diagnosis). "Unselected" or blank entries were omitted from score calculations.

3.3 System Overview

3.3.1 Image Preprocessing

Each panoramic `.png` image is read from disk, then base64–encoded in-memory and inlined into the `.json` payload as a `"data:image/png;base64` URI. This allows the GPT-4o Vision API to treat it exactly like any externally hosted image URI without requiring an extra hosting step.

3.3.2 Prompt Assembly

Every call to GPT-4o Vision is produced by filling three *run-time parameters* into a fixed prompt template:

- Questions with Answers ($\mathcal{Q}$):
  The list of 14 diagnostic questions with the associated choices from Table 3.2;
- Examples (ξ):
  An optional few-shot snippet containing 2 previously solved cases (used in all but the zero-shot setting);
- Image Path (p):
  The image relative path on disk.

These three variables are substituted into the `message` template and sent to the `chat completions` endpoint. To guard against occasional empty or malformed replies, the request is executed inside a retry loop that repeats until a non-empty response is generated. The pipeline is illustrated in the following pseudo code Algorithm 1.

TABLE 3.1: **Answer options for questions 1–14**

Question	**Answer Choices**
Q1: Which jaw contains the lesion?	• Mandible • Maxilla • Mandible and maxilla
Q2: The lesion center is in what region?	• Molar region • Ramus region • Incisor region • Sinus region • TMJ region • Canine/Premolar region
Q3: The relationship of the lesion to teeth is	• Apex associated Vital tooth • Apex associated Non-vital tooth • Apex associated tooth with unknown vitality • Root associated • Crown associated • Missing tooth associated • Not tooth associated
Q4: Please estimate the number of lesions	• 1 • 2 • ≥3 • Generalised lesion
Q5: What is the maximum size of the lesion?	• <2 cm • 2–3 cm • >3 cm
Q6: Where is the origin of the lesion?	• Central • Peripheral
Q7: The borders of the lesion are	• Corticated • Defined but not corticated • Diffuse
Q8: The loculation of the lesion is	• Unilocular • Multilocular • Not loculated
Q9: The contents of the lesions are	• Radiolucent • Radio-opaque • Mixed • Radiolucent with flecks • Opaque
Q10: Does the lesion contain one or more teeth?	• Yes • No
Q11: Does the lesion expand the bony cortex?	• Yes • No
Q12: Does the lesion cause root resorption?	• Yes • No
Q13: Does the lesion cause tooth displacement or impaction?	• Yes • No
Q14: What is the most possible lesion given the above information?	Free-form response

TABLE 3.2: Diagnostic questions and their predefined answer choices.

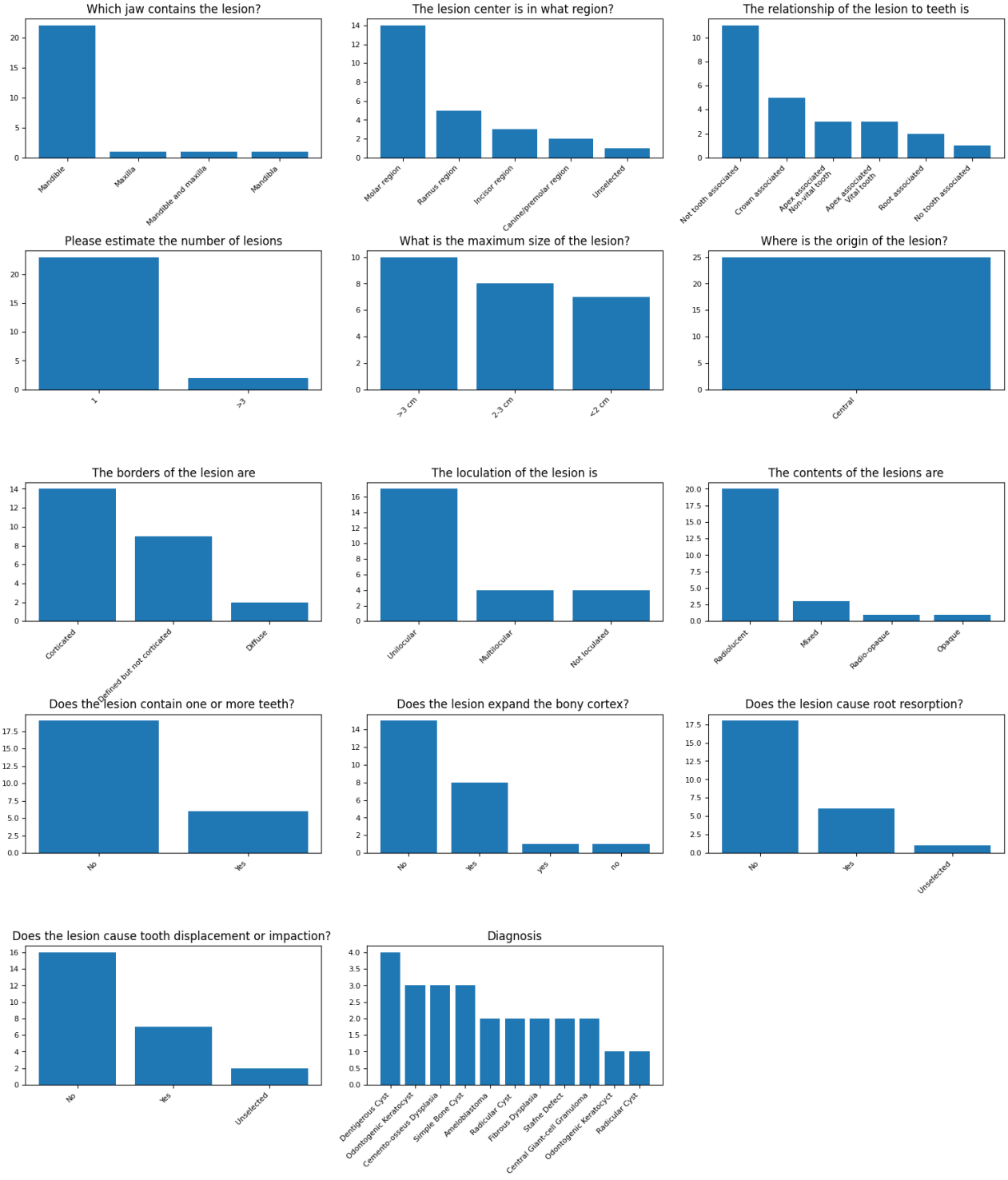

Figure 3.3: Answer distributions for the fourteen diagnostic questions. Most questions exhibit significant skew, which must be accounted for in the evaluation.

Algorithm 1: Baseline zero-shot annotation generation with ChatGPT-4o

Input: Image file path p, list of questions $\mathcal{Q}$ (with allowed options)
Output: Model response $\mathcal{A}$

FUNCTION *BaselineAnnGen*$(p, \mathcal{Q})$**:**

Encode image
open image at path p;
read bytes $\rightarrow$ base64 string b;

Compose chat messages
$\mathcal{M} \leftarrow$ {*system*: expert radiology instructions};
append *user* message containing:
– placeholder sentence ("Below is the image…")
– IMAGE_URL: `"` + b
– prompt with formatted question list $\mathcal{Q}$

Call OpenAI API
repeat
send $\mathcal{M}$ to `chatgpt-4o` with `max_tokens`=5000;
$\mathcal{A} \leftarrow$ first choice content
until $\mathcal{A} \neq \emptyset$
return $\mathcal{A}$

3.3.3 Post-processing

GPT-4o Vision is initially prompted (with the image, questions, and answer choices) to produce structured JSON directly. However, minor deviations, such as typos, extra commentary, or formatting quirks, can still cause strict parsing to break. To both enforce valid output and enable richer reasoning, the thesis adopts a two-stage pipeline, as illustrated in Figure 3.4.

1. **Generation agent.**
   GPT-4o Vision is prompted with each question and its set of allowed answer choices and instructed to select from among those options (also with intermediate "chain-of-thought" reasoning steps in certain related settings). This constrained format encourages the model to adhere strictly to the provided schema, producing more reliably structured outputs. The thesis also evaluated an alternative "free-form" approach, where the model first generates an unconstrained response and then maps it to the nearest choice. However, this approach consistently underperformed relative to the direct-choice setup (see Experiment 4.2.4).

2. **Extraction agent.**
   A GPT-4o-mini instance ingests the generation agent's output plus the original question list and choices. It corrects any malformed entries, normalizes near-matches (e.g. "mandibla" $\rightarrow$ "Mandible"), and emits clean, parseable JSON. The prompt of the extraction agent is illustrated in Appendix B.13.

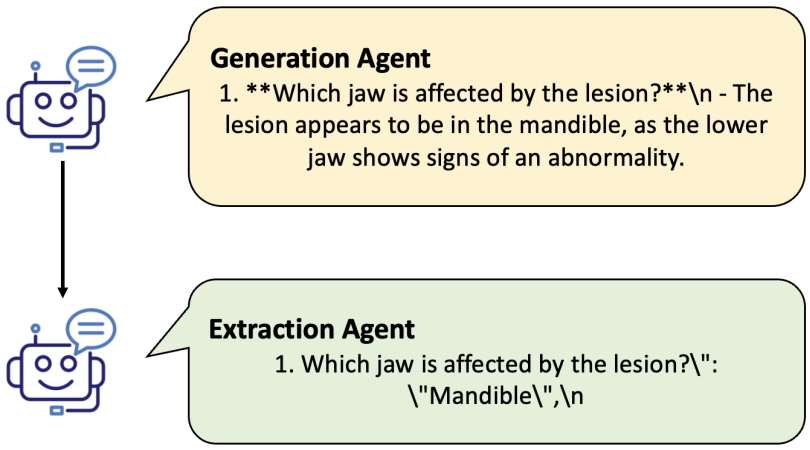

Figure 3.4: The two-agent workflow for generating and extracting answers.

This flow preserves GPT-4o's full explanatory power while guaranteeing a uniform, machine-readable output for evaluation. The 2-stage working pipeline with a constrained generation agent is illustrated in Algorithm 2.

Algorithm 2: Dual-agent pipeline for structured case annotation

Input: Image path p, question set $\mathcal{Q}$ (with allowed options)
Output: JSON string $\mathcal{J}$ containing 14 `question:answer` pairs

Function *ExtractAnswers*$(p, \mathcal{Q})$**:**

1. Encode image
open file p; read bytes $\rightarrow$ base64 string b
2. Agent 1 – Vision annotation
$raw \leftarrow$ BaselineAnnGen$(p, \mathcal{Q})$
3. Agent 2 – Information extraction
construct message list $\mathcal{M}$ with:
system 1 = JSON-only extraction guidelines + $\mathcal{Q}$;
system 2 = required output format (14 numbered keys);
user = "Below is the text to summarize" + *raw*
$\mathcal{J} \leftarrow$ `CallGPT(`*model* = *gpt-4o-mini*, $\mathcal{M}$`)`
return $\mathcal{J}$

3.4 Experiments Design

The experiments the thesis performed and the configuration of GPT-4o Vision the thesis used to comprehensively evaluate GPT-4o Vision are described as following.

3.4.1 Preliminary: GPT-4o Vision Hyperparameter Configuration

GPT-4o Vision exposes two primary sampling parameters: `temperature` and `top_p`. Setting `temperature = 0` forces greedy decoding, i.e., the model always selects the token with the highest probability, whereas higher `temperature` values introduce stochasticity by sampling from the full softmax distribution. The `top_p` parameter (nucleus sampling) constrains each token choice to the smallest possible set of candidates whose cumulative probability mass exceeds `p`; for example, `top_p = 1` imposes no filtering, allowing all tokens to be considered.

To minimize randomness and ensure reproducible results, the thesis fixes `temperature = 0` and `top_p = 0` for all experiments. Actually, after setting `temperature = 0`, the value of `top_p` no longer matters, as the zero temperature has already forced the model to always choose the next token with the highest probability. All model queries were run before May 22, 2025; any subsequent version updates may yield different outputs.

3.4.2 Model Performance Comparison

First, the thesis compares the performance of three state-of-the-art vision LLMs: Claude-Opus-4 [63], Grok-2-Vision-Latest [64], and GPT-4o-Vision [65], to select the best candidate for subsequent experiments. The thesis evaluates each model using per-question weighted recall and the overall average weighted recall (as defined in Section 3.5.1). This experiment is used to select the model for the evaluation part. As a result, GPT-4o Vision is the best and used for later experiments.

3.4.3 Baseline Performance

Next, the thesis establishes two baselines for GPT-4o Vision. First, a majority-class predictor is evaluated, which simply selects the most frequent label for each question, highlighting the impact of class imbalance. Second, GPT-4o Vision's zero-shot performance is measured, including its variance across multiple runs. Finally, the thesis tests whether allowing the model to generate free-form responses (rather than forcing it to choose from predefined options) affects its weighted recall.

1. **Majority Class Baseline**
   The thesis firstly establishes a simple baseline by assigning, for each question, the single most frequent (majority) label. As will be described in Section 3.5.1, the weighted recall for this baseline equals the fraction of examples in the largest class, directly reflecting each question's class imbalance. This baseline reveals whether GPT-4o Vision actually learns to interpret the image or merely defaults to majority-class guessing.

2. **Zero-Shot**
   To assess GPT-4o Vision's capabilities, all 14 questions (and answer choices)

are presented using only the base prompt template (no examples, chain-of-thought prompts, or additional cues). The thesis then computes weighted recall to compensate for severe class imbalances. This experiment quantifies the raw performance of GPT-4o Vision on panoramic image annotation without any prompt engineering. The prompt is offered in Appendix B.1

3. **Zero-Shot Variance**
   The thesis then repeats the zero-shot setup five times to measure the mean and variance of weighted recall across all questions. By evaluating consistency over multiple runs, it helps determine how many repetitions are needed in subsequent experiments to balance budget constraints against evaluation reliability.

4. **Zero-Shot vs. Free-Form Zero-Shot**
   In standard zero-shot prompting, GPT-4o Vision must select one of the provided answer options directly. Here, the thesis instead allows the model to generate an unconstrained, free-form response before mapping it to the closest choice (an approach called “zero-shot-free"). Comparing these two variants shows whether granting the model greater expressive freedom improves or degrades annotation accuracy. The prompt is offerend in Appendix B.2

3.4.4 Prompt Engineering Experiments

This thesis begins by surveying a variety of prompt-engineering strategies and selects the highest-performing approach based on validation score, measured by weighted recall. Using this optimized prompt, the thesis then benchmarks GPT-4o Vision across two usage scenarios, namely "leave-one-out", and "error-correction", together with ablation study, to rigorously assess its annotation robustness and reliability in the later sections.

The following prompt is used as a basic prompt template, based on which other prompt engineering techniques are performed.

Listing 3.1: Base prompt template sent to the OpenAI API

```
messages = [
    {
        "role": "system",
        "content": (
            "You are an oral radiology expert assistant. "
            "Analyze the oral panoramic image thoroughly. "
            "Then, for each numbered question, give a concise final answer
            drawn from the listed options. "
            "Do not say "I can't analyze images.""
            "This is for exploratory/educational use only."
        )
    },
    {
```

```
        "role": "user",
        "content": [
            {
                "type": "text",
                "text": (
                    """
                    Below is the panoramic image that you are going to help
                    annotate.
                    """
                )
            },
            {
                "type": "image_url",
                "image_url": {
                    "url": "<BASE64_IMAGE>"
                }
            },
            {
                "type": "text",
                "text": (
                    "Answer each question by number, selecting from the
                    provided choices."
                    "For the final question, provide your own
                    free-form response."
                    f"{questions}\n\n"
                    "Finally, list each question number followed by
                    your answer."
                )
            }
        ]
    }
]
```

The following prompt engineering techniques are explored.

1. **Zero-Shot vs. Few-Shot:**
   This experiment compares GPT-4o Vision's performance under zero-shot prompting with its performance under few-shot prompting. For the few-shot condition, two (question, answer) pairs are synthesized by running zero-shot prompts on representative images and selecting correct outputs. While this bootstrapping avoids manual annotation, it may propagate zero-shot biases and fail to capture the full anatomical variation. This experiment tests whether providing in-context examples improves annotation accuracy. The prompt for few-shot is provided in Appendix B.3.

2. **Few-Shot vs. Few-Shot + Images:**
   Two few-shot variants are evaluated: (1) textual examples only and (2) the same textual examples augmented with their corresponding panoramic slices. By comparing these settings, the effect of including images alongside text

examples on model performance is assessed. The prompt for few-shot with images is provided in Appendix B.4.

3. **Chain-of-Thought (CoT) with Few-Shot Prompting:**
   In this setup, GPT-4o Vision is instructed to generate intermediate, step-by-step reasoning prior to providing each answer. Two worked CoT examples (derived from zero-shot CoT) are included in the prompt to illustrate the desired reasoning depth and format. This experiment examines whether guiding the model to "think aloud" enhances diagnostic accuracy. The prompt is provided in Appendix B.5.

4. **Tree-of-Thought (ToT) with Few-Shot:**
   A beam-search-based ToT strategy is implemented, inspired by Yao et al. [61]. For each question, GPT-4o Vision first generates up to five candidate reasoning steps; a secondary GPT-4o Vision instance then scores each candidate on a 0–1 scale. Only the top three branches are retained for further expansion, with lower-scoring or duplicate-score branches pruned in favor of earlier entries. This experiment evaluates whether exploring multiple reasoning paths and backtracking yields more accurate answers than a single CoT. Detailed algorithms are illustrated in Algorithms 3, 4 and 5. The prompts are provided in Appendix B.6 and B.7.

Algorithm 3: ScoreBranch

```
// Give a score to an answer according to the image and
question
```

Input: branch text b, image (Base64) I, question q
Output: confidence score $s \in [0, 1]$
Construct a system+user prompt embedding b, I, and q
repeat
  Call `OpenAI.chat()` with model=`gpt-4o`; store reply s
until model replies with a non-empty, float number
return s

5. **Self-Consistency with Few-Shot:**
   The few-shot CoT prompt is executed five times (identical settings except for random seeds), and the five label predictions are collected. The final answer is chosen by majority vote. By aggregating multiple independent reasoning traces, this experiment assesses whether consensus among runs reduces random errors and improves reliability.

6. **Self-Debate with Few-Shot:**
   In this variant, GPT-4o Vision is prompted to adopt three personas—"Doctor A," "Doctor B," and "Doctor C", that each producing an independent reasoning chain and answer for each question. A follow-up instruction then directs

Algorithm 4: AskModel
`// Generate K candidate answers to a question, given the image and previous answers`

Input: image I, full question list Q, prefix answers $\mathbf{a}_{1:t-1}$, `previous_answers` string, branch width K, few-shot examples E
Output: set $\mathcal{C}$ of up to K candidate pairs $(\text{answer}, \text{score})$
Let $q_t \leftarrow Q[t]$ be the next unanswered question
Compose a chain-of-thought prompt containing E, I, $\mathbf{a}_{1:t-1}$ and q_t
Call `OpenAI.chat()` with $n = K$ to obtain K answer strings $\{b_i\}$
foreach b_i **do**
  $\sigma_i \leftarrow \text{ScoreBranch}(b_i, I, q_t)$
  Add (b_i, σ_i) to $\mathcal{C}$
return $\mathcal{C}$

Algorithm 5: ToTFewshots
`// Tree-of-thought with Beam Search`

Input: image path P, question list Q (length m), beam width B, branch width K, examples E
Output: best answer list $\mathbf{a}^\star$ of length m
Load and Base64-encode the image from $P \rightarrow I$
$\mathcal{B} \leftarrow \{(0, \langle\rangle)\}$ `// min-heap of (score, prefix)`
for $t = 1$ **to** m **do**
  $\mathcal{B}' \leftarrow \emptyset$
  foreach $(S, \mathbf{a}_{1:t-1}) \in \mathcal{B}$ **do**
    Build `previous_answers` string from $\mathbf{a}_{1:t-1}$
    $\mathcal{C} \leftarrow \text{AskModel}(I, Q, \mathbf{a}_{1:t-1}, \texttt{previous\_answers}, K, E)$
    foreach $(b, \sigma) \in \mathcal{C}$ **do**
      $\mathbf{a}_{1:t} \leftarrow \mathbf{a}_{1:t-1} \cup \{b\}$
      $S' \leftarrow S + \sigma$ `// cumulative score`
      Push $(S', \mathbf{a}_{1:t})$ into $\mathcal{B}'$
      if $|\mathcal{B}'| > B$ **then**
        pop worst element from $\mathcal{B}'$
  $\mathcal{B} \leftarrow \mathcal{B}'$
Take $(\_, \mathbf{a}^\star)$ with largest score from $\mathcal{B}$
return $\mathbf{a}^\star$

the model to reconcile any conflicts into a single, unified response. This experiment investigates whether generating diverse viewpoints and merging them improves decision-making. The prompt is provided in Appendix B.8.

7. **Self-Critique with Few-Shot:**
   Self-critique demonstrations are generated by running zero-shot CoT with an added critique step, so that each example shows an initial reasoning trace, a critical review highlighting ambiguities or errors, and a corrected final answer. At inference, GPT-4o Vision first produces its CoT answer and then receives this prompt:

   > IMPORTANT: For each question, you will:
   > 1) Provide your initial answer with a step-by-step chain-of-thought.
   > 2) Critically review that answer, identifying any weaknesses or alternative interpretations. 3) Offer a final, revised answer based on your self-critique.

   This experiment evaluates whether explicitly asking the model to critique its own reasoning reduces hallucinations and improves final accuracy. The prompt is provided in Appendix B.9.

3.4.5 Hyperparameter and Sequence Effects

This section examines two potential sources of variability in GPT-4o Vision's performance. First, the impact of different `temperature` settings is evaluated to see if adding controlled randomness affects diagnostic accuracy. Second, the order of the first thirteen questions is randomized (with the diagnosis question always last) to determine whether sequencing introduces hidden dependencies that alter the quality of answers.

For each experiment, five temperature values and five question-order permutations are tested while all other settings remain constant. Performance is measured by weighted recall, and statistical significance is evaluated using the Friedman test.

3.4.6 Usage Scenarios

Two practical scenarios evaluate how GPT-4o Vision can leverage supplemental information to improve diagnostic accuracy and self-correct when mistakes occur.

1. **Leave-One-Out**
   For each of the fourteen questions, the model is given the ground-truth answers to the other thirteen and asked to predict only the withheld label using CoT with few-shot prompting. This procedure is repeated across all 25 cases, and the weighted recall for each question is recorded. Such a setup mimics a clinical workflow in which most findings are already documented, and the model must infer a single missing item from both textual context and visual input. The prompt is provided in Appendix B.10.

2. **Correcting a Single Unknown Mistake**
   Here, GPT-4o Vision is provided with a complete set of fourteen answers, one of which is deliberately incorrect (the model is not told which). Using the same CoT with a few-shot prompt, it must first identify the erroneous response and then replace it with the correct label. This scenario simulates a tutoring or peer-review use case, where the model helps to locate and correct a single mistake in a student's diagnostic report. The prompt is provided in Appendix B.12.

3.4.7 Ablation Study

1. **Answering a Single Question at a Time**
   In this experiment, GPT-4o Vision was prompted to answer each question individually using CoT with few-shot prompting rather than receiving the full sequence of fourteen questions. Thus, each prompt contained exactly one question (including Q14, the diagnosis), with no prior answers provided as context. By eliminating any interdependencies, this design forces the model to rely solely on visual information for each feature. It reveals how much the answers depend on contextual cues versus pure image interpretation.

2. **Leave-One-Out without Images**
   This variant repeats the leave-one-out protocol but removes the input image entirely. For each question, the model is given the correct labels for the other thirteen and must predict only the withheld label based solely on textual context. Comparing weighted recall with and without the image demonstrates whether GPT-4o Vision genuinely leverages visual data or primarily infers answers from surrounding text. The prompt is provided in Appendix B.11.

3.5 Metrics

In this section, metrics used to evaluate models are introduced. For evaluating most of the model performance, weighted recall per question is used to analyze the model's performance across each question. In experiments investigating whether the temperature and question order can affect performance, the Friedman test is used to test against the null hypothesis H_0. Finally, in the experiment evaluating GPT-4o Vision's ability to spot and correct wrong answers, accuracy is assessed across each question.

3.5.1 Weighted Recall

The weighted recall is defined as the following:

Recall is defined per class c as

$$\text{Recall}_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c},$$

and the *weighted recall* across all classes is

$$\mathrm{WR} = \sum_{c \in C} \underbrace{\frac{N_c}{N}}_{w_c} \mathrm{Recall}_c,$$

where N_c is the number of true examples of class c and $N = \sum_c N_c$. A majority-class predictor has $\mathrm{Recall}_{c^*} = 1$ for the dominant class c^* and $\mathrm{Recall}_{c \neq c^*} = 0$, so its overall weighted recall reduces to

$$\mathrm{WR}_{\mathrm{majority}} \;=\; \frac{N_{c^*}}{N},$$

i.e. exactly the fraction of examples in the largest class.

Because each class's weight $w_c = N_c/N$ grows with its frequency, common labels contribute more to the final score than rare ones. This matches the real-world expectation that performance on frequently encountered cases should carry greater emphasis, while errors on very rare labels, such as cases that even clinicians might struggle to diagnose from images alone, are penalized less severely. In practice, a more nuanced weighting could be obtained by consulting domain experts so that each label's weight reflects not only its frequency but also its clinical significance.

When determining whether a generated answer should be considered correct, the thesis cannot rely on exact string matching alone as small variations in word order, capitalization, or minor typos may not change the underlying meaning but would nonetheless be marked as incorrect. To address this, sparse vector embeddings are used to encode both the ground-truth string and the model's prediction. Then, the cosine similarity between these two embeddings is computed. If the similarity exceeds a threshold of 0.80, the prediction is treated as a correct match. Sparse embeddings effectively capture the presence of key terms while ignoring small perturbations; minor noise in spelling or formatting will only slightly lower the cosine score, allowing semantically equivalent answers to register as matches rather than false negatives.

3.5.2 Statistical Testing for Temperature and Order Effects

To determine whether varying `temperature` or shuffling question order significantly affects GPT-4o Vision's weighted-recall performance, this thesis employs the Friedman test. Each of the 14 diagnostic questions serves as a block whose weighted recall is measured under k=5 conditions (five temperatures or five orderings). Within each block, the model's scores are ranked from 1 to k (with tied scores assigned average ranks). Denote by $R_{i,j}$ the rank of condition j on question i. The Friedman statistic is then

$$\chi^2_F \;=\; \frac{12}{n\,k\,(k+1)} \sum_{j=1}^{k} \Big(\sum_{i=1}^{n} R_{i,j} \Big)^2 \;-\; 3\,n\,(k+1),$$

where $n = 14$ is the number of questions. Under the null hypothesis (that all k conditions yield equivalent performance), χ_F^2 approximately follows a chi-square distribution with $F = k-1 = 4$ degrees of freedom (as there are 5 experiments each, therefore $k = 5$.) A p-value below $\alpha = 0.05$ indicates that at least one condition produces a systematically different weighted recall.

The Friedman test was selected because it compares each question's performance across multiple conditions (temperatures or orderings) as related samples without assuming a normal distribution. By treating each question as a "block" and ranking the five weighted-recall scores within that block, it effectively removes question-to-question variability and isolates only the impact of changing temperature or changing question order. In both the temperature and order experiments, all other prompt settings remain fixed while only the five conditions vary, so the Friedman statistic directly evaluates whether those condition changes produce a consistent shift in performance across questions. Finally, a 5% significance level is used because it represents the conventional balance between Type I and Type II error risks in empirical research.

Chapter 4

Experiments and Results

This chapter presents a series of experiments designed to assess GPT-4o Vision's ability to annotate oral panoramic radiographs under various prompt-engineering strategies. Firstly performances of GPT-4o Vision, Grok-2-Vision-Latest and Claude-Opus-4 are compared to select the best model for later evaluation. Then the thesis begins with a majority-class baseline that consistently predicts the most frequent label, highlighting class imbalance. This chapter then compares zero-shot, few-shot, chain-of-thought, tree-of-thought, self-consistency, self-debate, and self-critique prompts to identify the best-performing approach. Using that prompt, two realistic scenarios are explored: (1) "leave-one-out," where the model infers a single missing answer given the other correct labels and the image, and (2) "correct-a-single-mistake," where the model must detect and fix one wrong label in a complete answer set. Finally, two ablation studies test the model's reliance on context versus vision: answering each question in isolation and repeating the leave-one-out process without any images.

All experiments (except "Correcting a Single Unknown Mistake", which reports accuracy for detection and correction) use weighted recall to address severe class imbalance. To test whether temperature or question order significantly affects performance, the Friedman test at a 5 % significance level is applied.

4.1 Model Performance Comparison

Figure 4.1 compares per-question weighted recall for GPT-4o Vision, Grok-2-Vision-Latest, and Claude-Opus-4. While all three models achieve near-identical performance on the highly imbalanced Q1, Q4, and Q6, substantial differences emerge on more challenging questions, most notably Q3 (Relationship to the Surrounding Teeth), Q7 (Lesion Contour), Q10 (Multiple Tooth Involvement), and Q12 (Root Resorption). In particular, GPT-4o Vision consistently outperforms its peers on Q3 and Q10, demonstrating superior sensitivity to these lower-frequency features. When averaged over all fourteen questions, the overall weighted-recall scores of the three models are listed in Table 4.1.

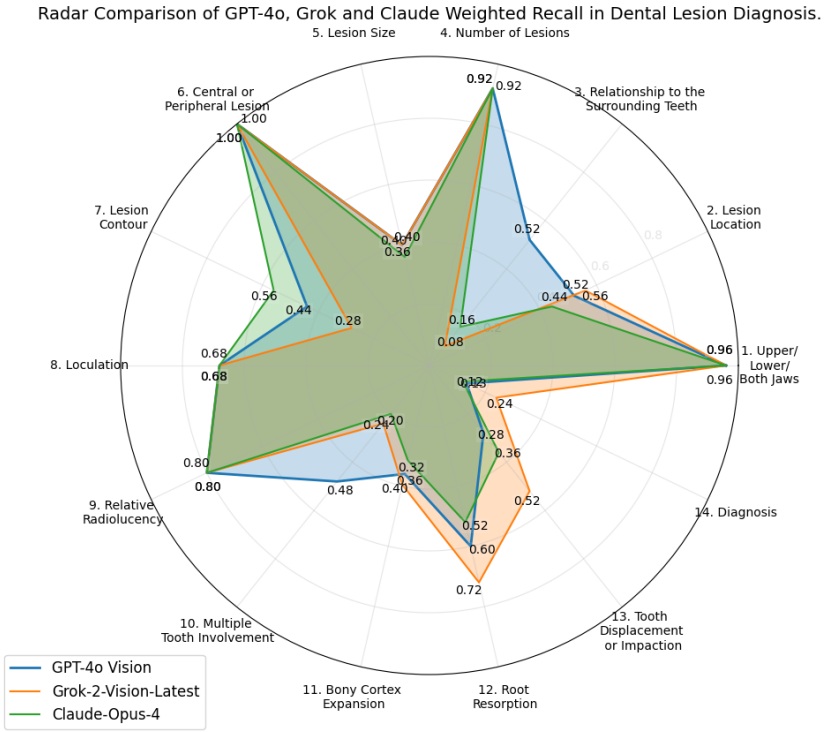

FIGURE 4.1: Weighted recall per question among GPT-4o Vision, Grok and Claude.

Model	**Average Weighted Recall**
GPT-4o Vision	0.578
Grok-2-Vision-Latest	0.557
Claude-Opus-4	0.528

TABLE 4.1: Average weighted-recall scores for each vision-LLM across all fourteen diagnostic questions.

Because GPT-4o Vision attains the highest average weighted recall (0.578), it is selected as the primary model for all subsequent experiments in this thesis.

4.2 Baseline Performance

4.2.1 Majority Class Baseline

The baseline model is established by using a model that, for each question, always predicts the single most frequent (majority) label. And as mentioned in Section 3.5.1, the weighted recall of this model is exactly the fraction of examples in the largest class, reflecting the imbalance of each class.

Figure 4.2 visualizes these values for each of our fourteen questions. The gray background then serves as a visual anchor in all subsequent radar plots, reminding how skewed the data are.

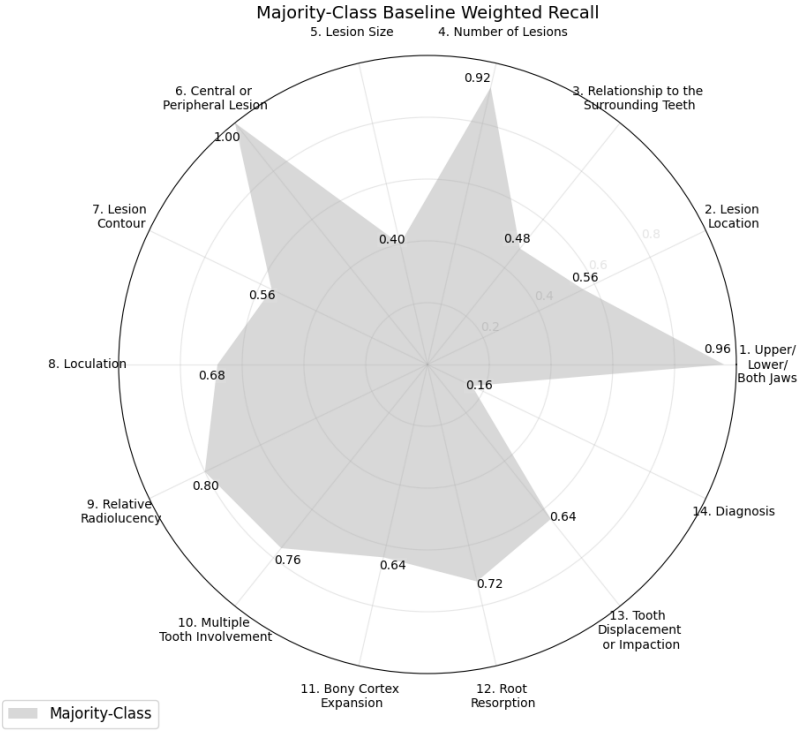

Figure 4.2: Weighted-recall of a majority-class predictor for each diagnostic question. By construction, these values coincide with the dominant class frequencies.

Key observations:

- Questions Q1 (Upper/lower/both jaws), Q4 (The number of lesions), and whether

it Q6 (Central or peripheral lesion) are extremely imbalanced (> 90% in a single class), so even a predictor simply picking the most common answer scores nearly 1.0.
- Questions Q2 (Lesion location), Q3(Relationship to the surrounding teeth), and Q5 (Lesion size) have a majority class around only 40–50%, so they present a real opportunity for any model to demonstrate genuine understanding.
- Mid-range imbalance appears in other questions around 56–68%.

Conclusion:
This baseline indicates that any model whose per-question weighted recall falls below the shade is effectively no better than guessing the most common answer.

4.2.2 Zero-Shot

Figure 4.3 overlays the zero-shot weighted recall curve (blue) on top of the majority-class baseline (gray). The "Diagnosis" axis (Question 14) is highlighted in red and bold since its answer is a free-form "best guess" derived from the first 13 feature predictions.

Key observations:
- For questions Q1, Q4, Q6, Q8, and Q9, zero-shot recall equals the majority baseline. Therefore, GPT-4o Vision simply predicts the dominant class and fails to capture rarer cases. Therefore, it indicates GPT-4o Vision is mainly relying on its prior knowledge instead of vision clues.
- On questions Q2, Q3, Q5, Q7 and Q10, zero-shot recall falls below the majority, indicating that current GPT-4o Vision's visual reasoning ability alone does not reliably identify these attributes.
- Despite feeding back the first 13 answers, the diagnosis question Q14 achieves recall near 0.13, which is very low, highlighting the difficulty of synthesizing correct pathology without domain-specific training.

Analysis
These results demonstrate that, without prompt engineering or domain-specific tuning, GPT-4o Vision's visual reasoning is unreliable. When a question's weighted recall matches the majority-class baseline, it suggests the model is defaulting to the most common answer from its training corpus rather than truly interpreting the image. Conversely, for questions where weighted recall falls below the majority baseline, two factors may be at play:

- The model may still be guessing the dominant label from its own (misaligned) corpus distribution, illustrated by Figure 4.4 for the "Tooth Displacement or Impaction" example.
- The underlying label distribution in GPT-4o Vision's training data may be relatively balanced, causing the model to hedge or become uncertain when predicting those features, which in turn drives down recall.

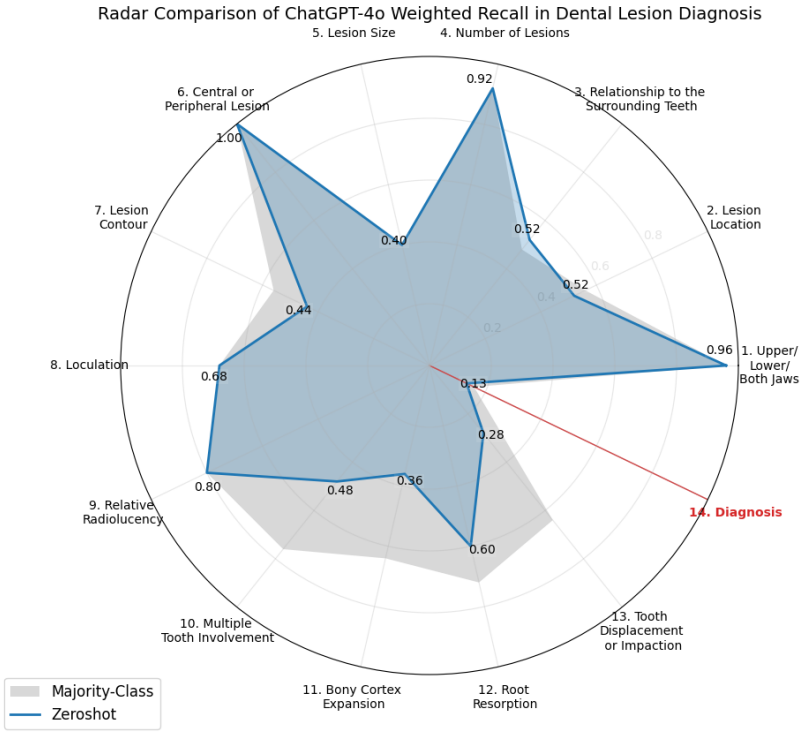

Figure 4.3: Weighted recall of GPT-4o Vision under zero-shot prompting (blue) vs. majority-class baseline (gray). Question 14 (Diagnosis) is shown in red to denote its free-form nature.

Conclusion
In either case, the low scores indicate that GPT-4o Vision struggles to extract nuanced radiographic features from panoramic images, relying more on textual priors than on its vision capabilities.

4.2.3 Zeroshot-Variance

Figure 4.5 plots, for each of the 14 diagnostic questions, the mean weighted-recall (orange polygon) and its variability (red error bars and colormap-shaded markers) over five independent zero-shot runs. Across 13 of the 14 questions, recall variance is negligible (all error bars nearly collapse), justifying the choice to run subsequent experiments only once under tight API budget constraints. Moreover, there is no clear relationship between class-support skew and recall variability, indicating that

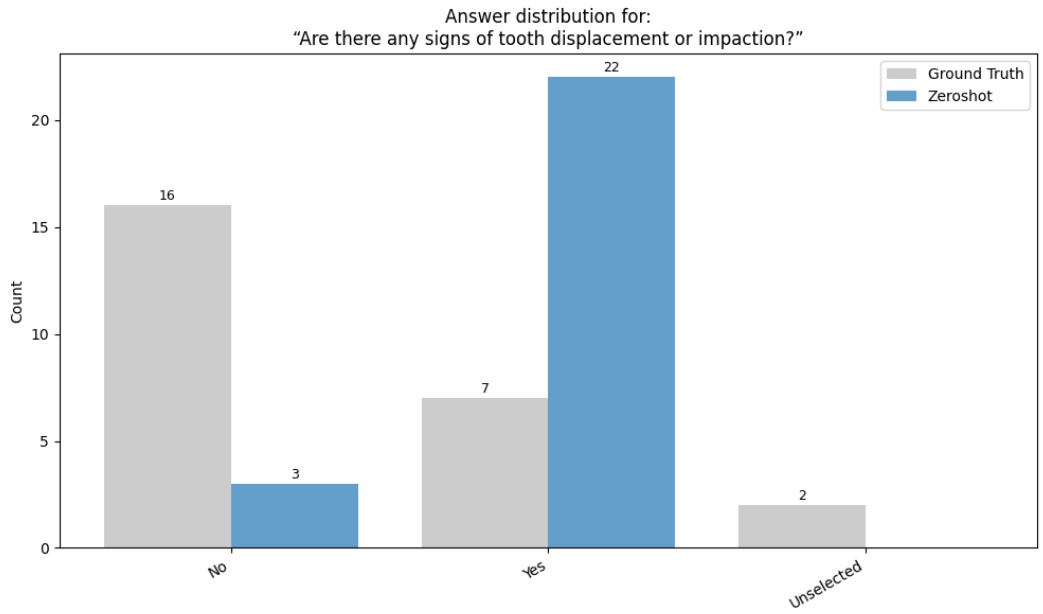

FIGURE 4.4: The figure compares GPT-4o Vision's selected answers against the majority-class baseline on Q13. It shows that GPT-4o Vision may default to the most frequent label in its training corpus even when that label does not match the true distribution of our dataset.

GPT-4o Vision's consistency is not simply a function of how imbalanced a question's labels are.

Analysis

Among all fourteen questions, only Q3 ("Relationship to the Surrounding Teeth") exhibited a notably high variance under zero-shot prompting. Upon inspecting Q3's answer set (Table 3.2), it became apparent that some options, such as "Not tooth associated" versus "Missing tooth associated", are semantically close and could easily confuse the model. To test this hypothesis, the prompt for Q3 was augmented by appending explicit definitions for each choice (e.g., clarifying what "Not tooth associated" means versus "Missing tooth associated"). Figure 4.6 shows the resulting mean weighted recall and variances.

After adding these definitions, Q3's variance dropped dramatically, confirming that ambiguous labels were driving its instability. However, this intervention also lowered Q3's average score (and unexpectedly depressed Q12's performance), suggesting that the model may have over-relied on textual information for Q3 at the expense of its visual reasoning or even disturbed latent dependencies between questions. Because the overall loss in accuracy outweighed the benefit of reduced vari-

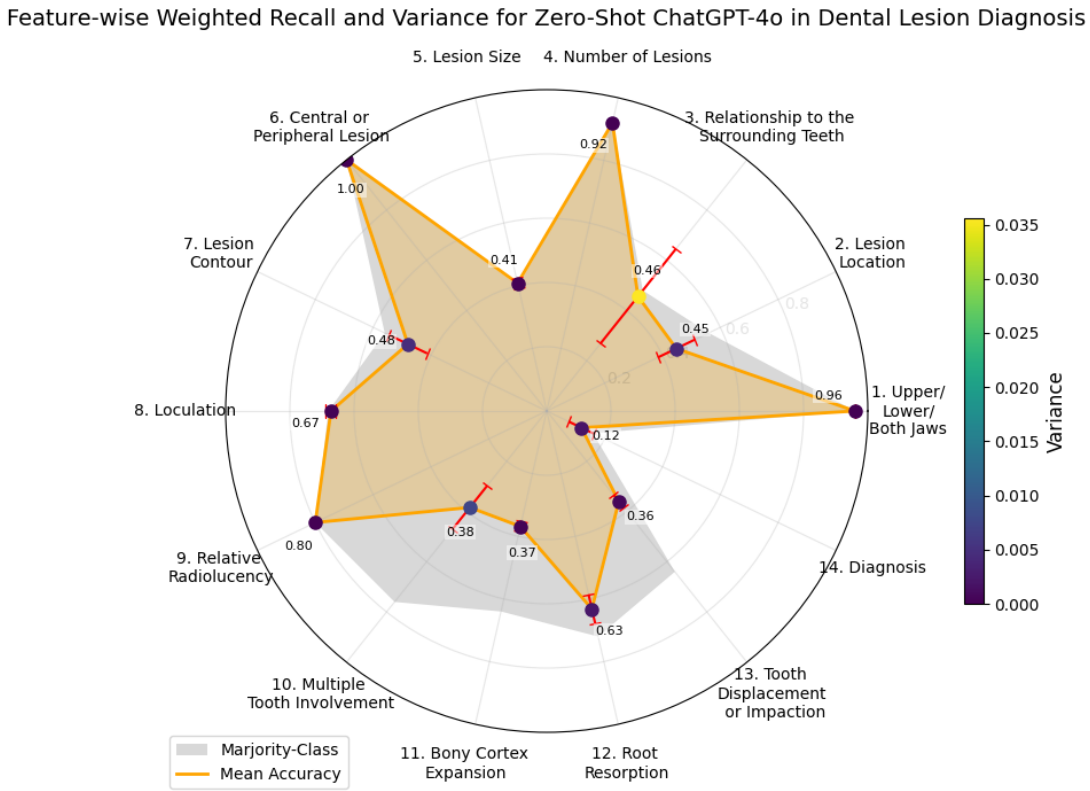

Figure 4.5: Zero-shot weighted-recall mean ± std across 14 questions (orange), overlaid on the majority-class baseline (gray). Error bars span ±1 std over five runs; marker color encodes per-question variance. GPT-4o Vision's recall is extremely stable for 13/14 features, supporting one-shot evaluations under budget limits, and variance does not track label skew.

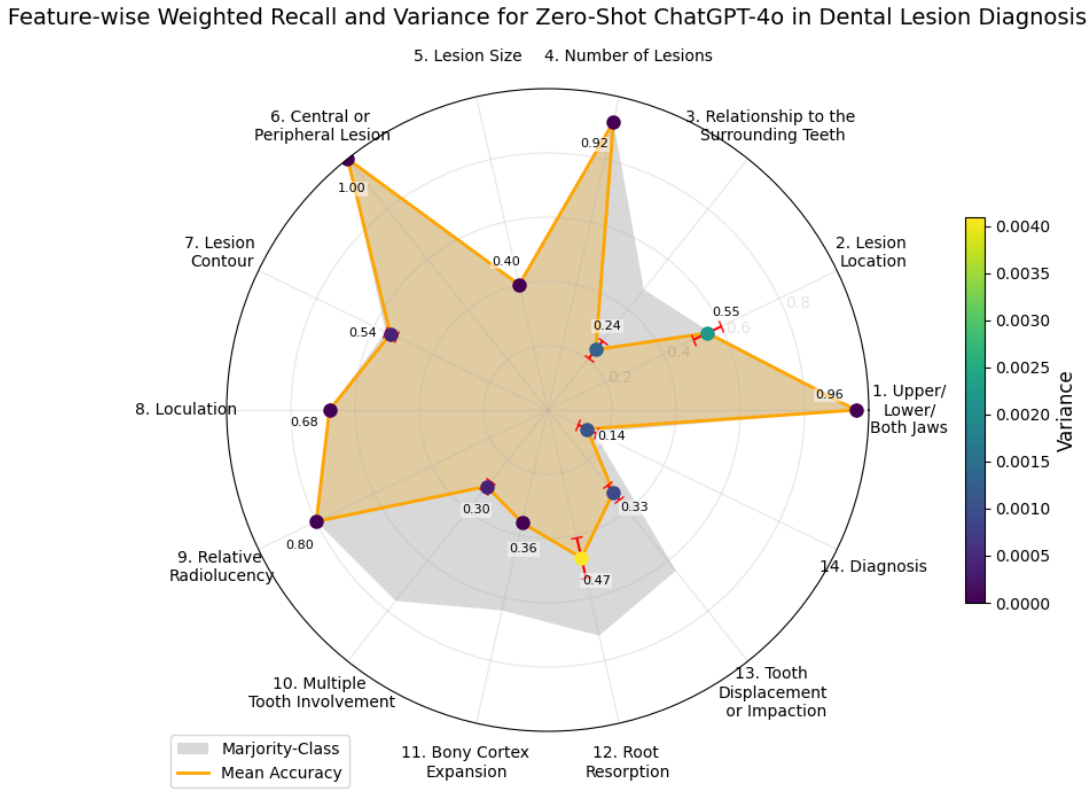

FIGURE 4.6: The mean weighted recall per question and variance per question after providing definitions of each choice of Q3

ance, and since thirteen of the fourteen questions already exhibited near-zero variance, this "definition augmentation" was not carried forward into later experiments.

Conclusion
Given that most questions demonstrated negligible run-to-run variability, and because of API budget constraints, the remaining experiments were conducted with a single pass per configuration. This choice preserves limited resources while maintaining confidence that the results would not materially differ if additional repetitions were performed.

4.2.4 Zero-Shot vs. Free-Form Zero-Shot

Next, the thesis relaxes the constraint that the generation agent must choose exclusively from the provided answer list, allowing it to elaborate freely before con-

verging on a final selection, which is called "zero-shot-free." Figure 4.7 overlays the two prompts' weighted-recall profiles (blue=zero-shot, orange=zero-shot-free) against the majority-class baseline.

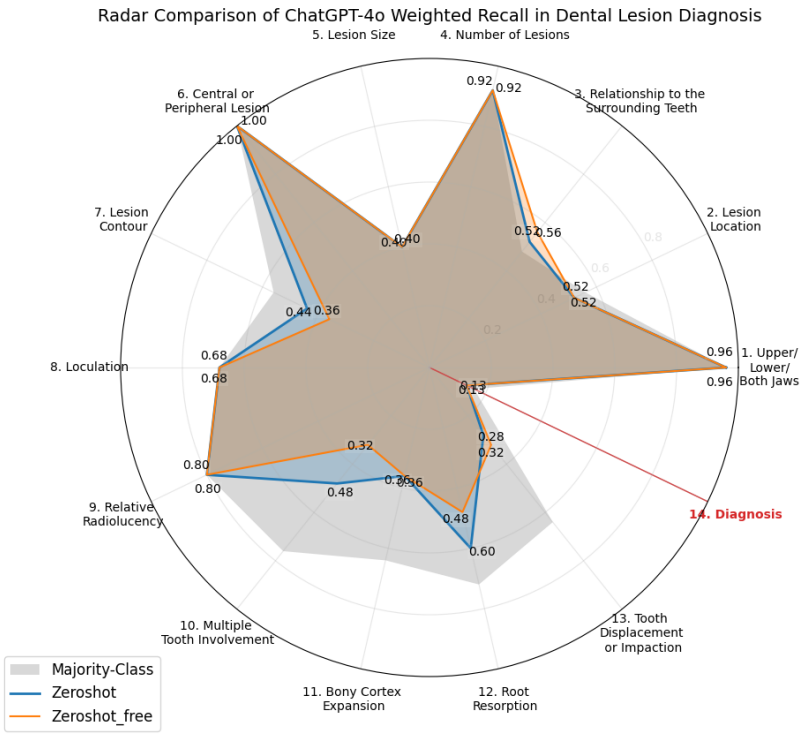

Figure 4.7: Comparison of weighted recall per question under standard zero-shot (blue) vs. free-form zero-shot (orange) prompting.

Conclusion

The overall mean weighted recall is 0.578 for zero-shot and 0.558 for zero-shot_free. Despite granting greater expressive latitude, the free-form prompt yields no measurable improvement and actually even shows a slight drop across most features. Moreover, the added variability in output format complicates downstream JSON extraction without boosting accuracy. Consequently, the standard zero-shot prompt is retained for all subsequent experiments.

4.3 Prompt Engineering Experiments

4.3.1 Zero-Shot vs. Few-Shot

Having established the zero-shot baseline (Fig. 4.3), the thesis next evaluates the impact of a simple few-shot prompt. Figure 4.8 compares the per-question weighted recall of standard zero-shot (blue) against few-shot (orange) prompting, with the majority-class baseline shown in gray.

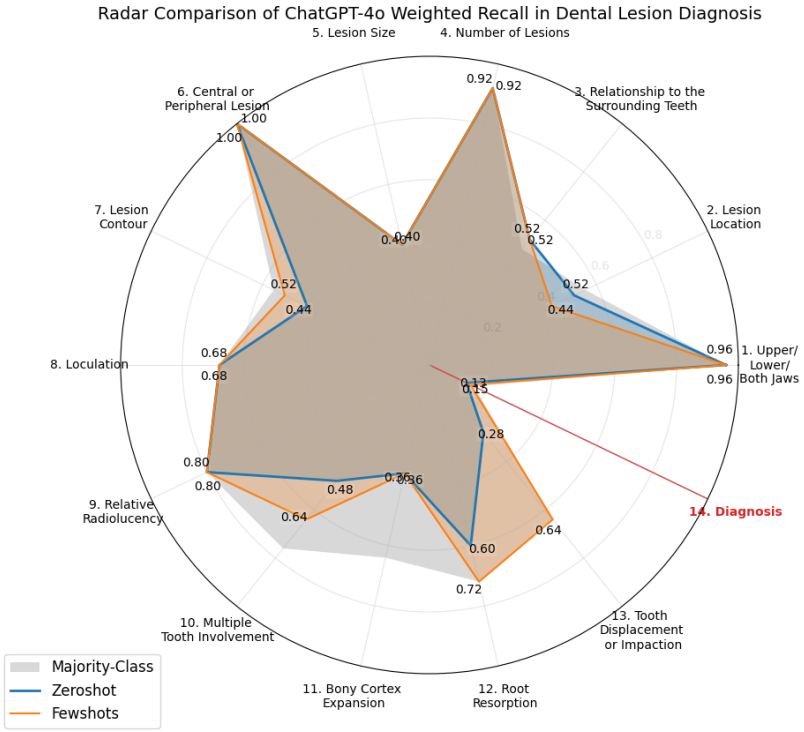

FIGURE 4.8: Weighted-recall per question under zero-shot (blue) vs. few-shot (orange) prompting. The gray area denotes the majority-class baseline. Diagnosis (Q14), derived from prior answers, is bolded in red.

Key observations

- Few-shot raises weighted-recall on Q10 and Q12, and Q13, bringing them up to the majority baseline.
- For features dominated by a single label (e.g., Q6), few-shot yields negligible changes, underscoring that few-shot cannot overcome extreme class skew.

- Question 14 continues to lag far below other features, confirming that even with examples, the model struggles to integrate all upstream inferences into a correct final diagnosis.

Analysis

Although only one "Yes" and one "No" example are provided in few-shot for Q13, exposure to both cases appears to help GPT-4o Vision resolve ambiguity in subsequent instances. Figure 4.9 shows the raw prediction counts for Q13. After few-shot, GPT-4o Vision outputs "Yes" far more often (10 → 15) and "No" less often (22 → 15), moving much closer to the ground-truth split (16 / 7 / 2). This demonstrates that even a small number of contrastive, domain-relevant examples can substantially correct over- or under-prediction of minority classes. However, the figure only reflects the label distribution instead of matching.

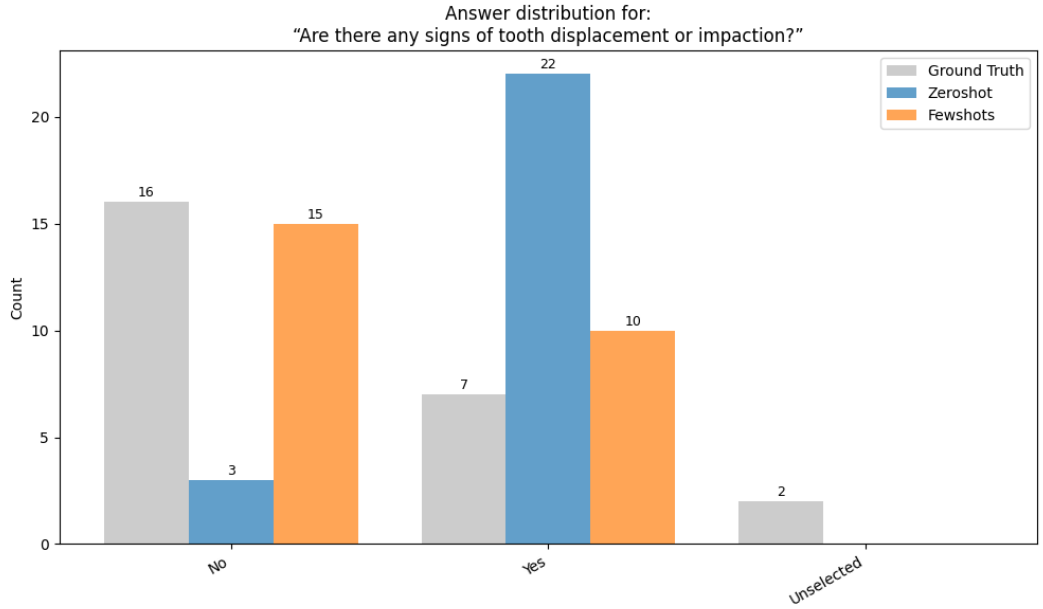

Figure 4.9: Answer-count distributions for Q13 ("Are there any signs of tooth displacement or impaction?") under zero-shot vs. few-shot. Ground truth (gray) is 16 No / 7 Yes / 2 Unselected; zero-shot (blue) overpredicts Yes (22) and underpredicts No (3); few-shot (orange) shifts toward the true distribution.

Conclusion

Few-shot prompting improved performance by presenting contrastive examples that encouraged consideration of all answer options rather than defaulting to the most frequent choice. The average weighted recall under zero-shot prompting was 0.578, compared to 0.625 with few-shot prompting. Consequently, few-shot prompting was selected to replace zero-shot prompting.

4.3.2 Few-Shot vs. Few-Shot + Images

In this experiment, two variants of few-shot prompting are compared: (1) standard few-shot (textual examples only) and (2) few-shot+images (the same textual examples accompanied by their corresponding CT slices). Figure 4.10 presents the per-question weighted-recall performance of GPT-4o Vision under these two conditions, with the majority-class baseline shown in gray and the diagnosis axis (Q14) highlighted in bold red.

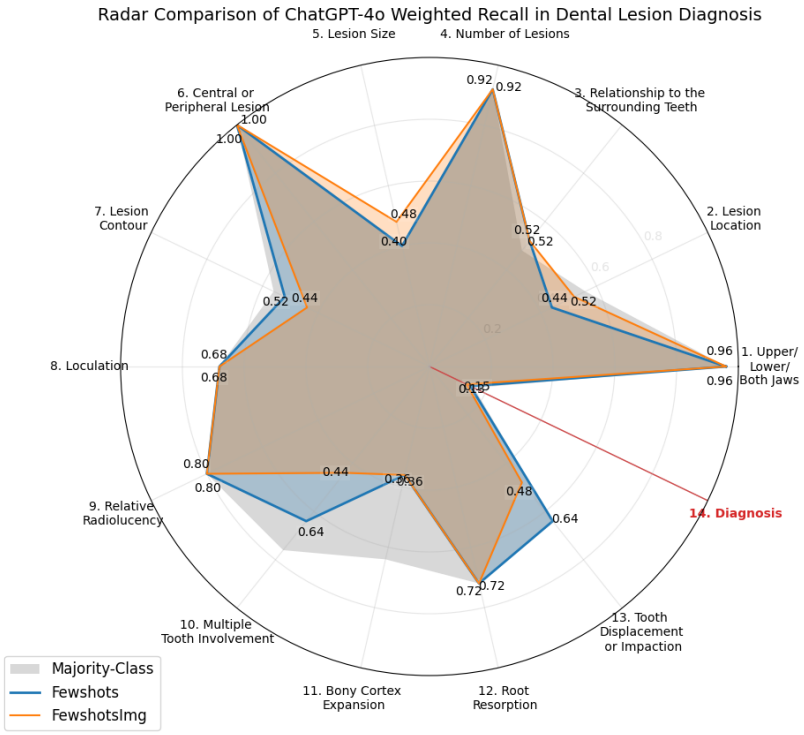

FIGURE 4.10: Per-question weighted-recall for standard few-shot (blue) versus few-shot+images (orange) prompting. The gray band denotes the majority-class baseline; Q14 (Diagnosis) is bolded in red because it is derived from prior feature answers.

Key observations:

- Overall, adding images to the few-shot examples did not yield any consistent improvement. Instead, performance on two features deteriorated.

- Q10 and Q13 saw their weighted recall drop by 4–8 points when the supporting images were included, suggesting that the visual context may have introduced additional ambiguity rather than clarification.
- All other questions remain essentially unchanged, indicating that for this structured annotation task, the textual examples alone suffice to guide the model's predictions.
- The diagnosis step (Q14) remains the most challenging, with both variants scoring well below the majority-baseline level.

Conclusion
Including images in few-shot reduces the performance by confusing the model, instead of improving the performance. Compared with few-shot whose average weighted score is 0.625, few-shot with images only reaches 0.569. Therefore, few-shot with images is not used.

4.3.3 Chain-of-Thought (CoT) with Few-Shot Prompting

After observing that few-shot examples alone yield a modest improvement over zero-shot, the thesis next explores combining few-shot demonstration with an explicit chain-of-thought (CoT) instruction. Figure 4.11 shows per-question weighted-recall under few-shot (blue) and chain-of-thought with few-shot (orange).

Key observations:

- Few-Shot achieves an average weighted-recall of 0.625. CoT + Few-Shot improves that to 0.642, a modest but consistent boost across most questions.
- Both Q10 and Q13 surpassed the majority baseline. In both cases, explicitly prompting the model to "think step by step" appears to help it recognize subtle, low-frequency patterns that Few-Shot alone misses.
- Even with CoT + Few-Shot, recall on the free-form diagnosis remains very low (0.15 vs. 0.13 under Few-Shot). This underscores that synthesizing the final pathological judgment remains extremely challenging without domain-specific fine-tuning or specialized training data.

Conclusion
Combining few-shot examples with an explicit chain-of-thought instruction consistently outperforms standard few-Shot, especially for questions where rare labels demand deeper reasoning. In particular, Q10 and Q13 cross their majority-baseline thresholds only under CoT + Few-Shot. Consequently, Chain-of-Thought with Few-Shot is adopted as the default prompt strategy from now on due to its average score 0.641 is higher than few-shot (0.625).

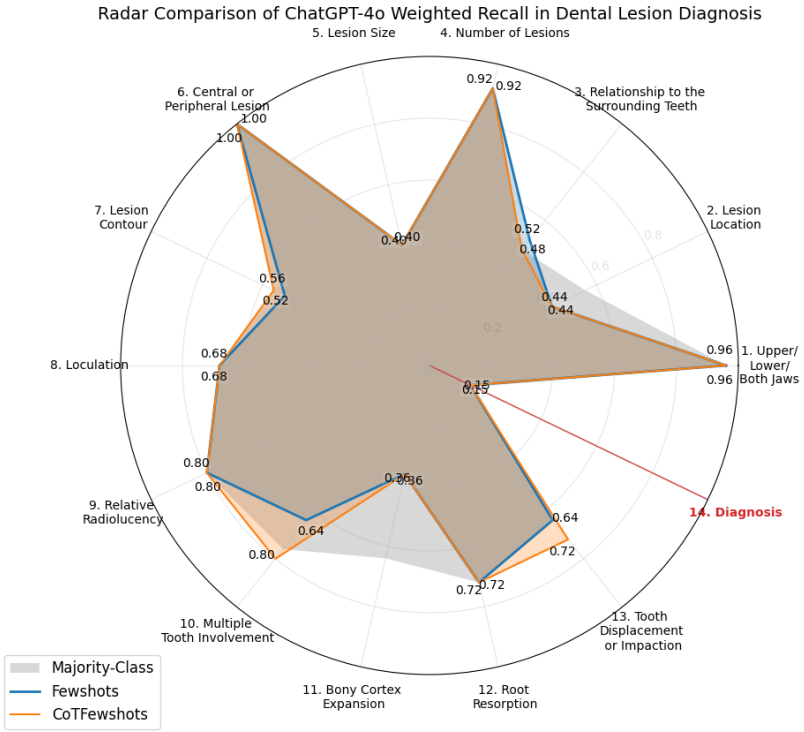

FIGURE 4.11: Per-question weighted-recall: Few-Shot (blue) vs. CoT + Few-Shot (orange). Gray shading indicates the majority-class baseline.

4.3.4 Tree-of-Thoughts (ToT) with Few-Shot

In ToT, the model does not commit to a single linear CoT narrative; instead, at each question it generates multiple candidate reasoning branches, uses a secondary evaluator to score them, and uses beam search to retain only the top-scoring branches. The evaluator used in this method is GPT-4o Vision itself. The goal is to recover from early missteps by exploring alternative sub-paths. Figure 4.12 compares the performances of CoT with few-shot with ToT with few-shot.

Key observations:

- Questions Q3, Q10 and Q13 saw great drop on the weighted score after using tree-of-thought.
- ToT's Q14 recall = 0.15 (unchanged from CoT), underscoring that even extensive branch-exploration cannot compensate for limited domain knowledge.

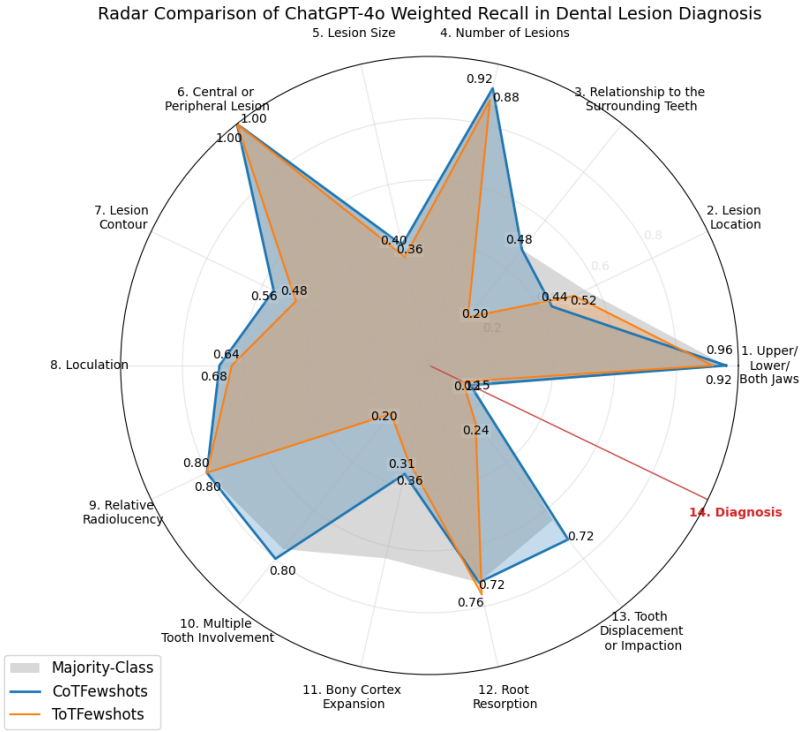

Figure 4.12: Per-question weighted-recall: CoT + Few-Shot (blue) vs. ToT + Few-Shot (orange). Gray area = majority-class baseline. Although ToT explores multiple reasoning branches, its overall recall drops relative to CoT, suggesting the self-evaluator misranks candidate chains.

Analysis

Since GPT-4o Vision is used both to generate candidate reasoning steps and to score them, any bias in its scoring function can steer the beam search toward superficially plausible yet erroneous branches. Figure 4.13 illustrates the search path for the first case across the first three questions; for simplicity, only the single top-scoring candidate at each step is shown. At each level, GPT-4o Vision produces three alternative answers with associated scores, yet the score assigned to the ground-truth response (highlighted in green) is lower than those of the incorrect candidates. This indicates that GPT-4o Vision's self-evaluation often overestimates incorrect but fluent outputs, suggesting that a more robust evaluation metric is required to guide the search effectively.

Conclusion

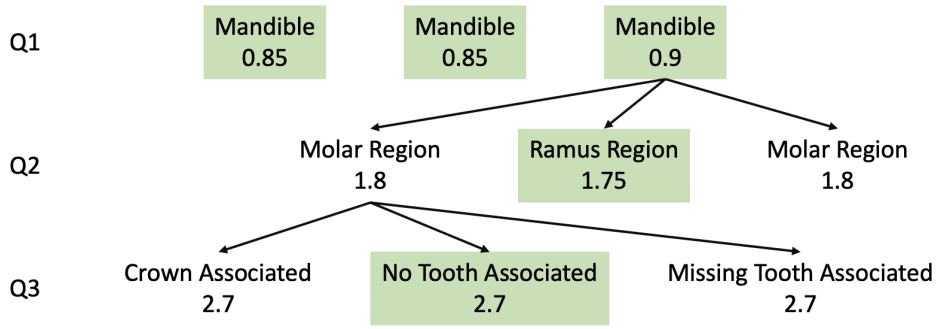

FIGURE 4.13: The search path of the first case for the first three questions performed by tree-of-thought. The green nodes are ground truth.

Tree-of-thought prompting cannot bring improvements for now. But it can be effective if a better evaluation approach is utilized. Because ToT + Few-Shot (recall = 0.531) underperforms CoT + Few-Shot (recall = 0.641), the thesis does not select ToT to replace CoT. Future work might investigate alternative evaluators (e.g., rule-based checks or human-curated heuristics) to improve branch selection to improve the overall performance of ToT.

4.3.5 Self-Consistency with Few-Shot

In this experiment, the thesis evaluates whether self-consistency can improve the chain-of-thought (CoT) performance under a few-shot setting. The CoT few-shot experiment (Section 4.3.3) is run five times per question, and the most frequent answer across those runs is selected as the final output. Example prompts for self-consistency are generated using a zero-shot agent. Figure 4.14 shows per-question weighted recall for: (1) CoT with few-shot alone (blue) and (2) CoT with few-shot enhanced by self-consistency (orange).

Conclusion
As the results indicate, incorporating self-consistency yields only a marginal change in performance, which is unsurprising given the low variance in individual CoT outputs. Specifically, the average weighted recall for CoT with few-shot is 0.641, compared to 0.635 for the self-consistent variant. Because this 0.006 difference falls within the expected generation variance and because self-consistency requires running the model multiple times, subsequent experiments do not employ self-consistency.

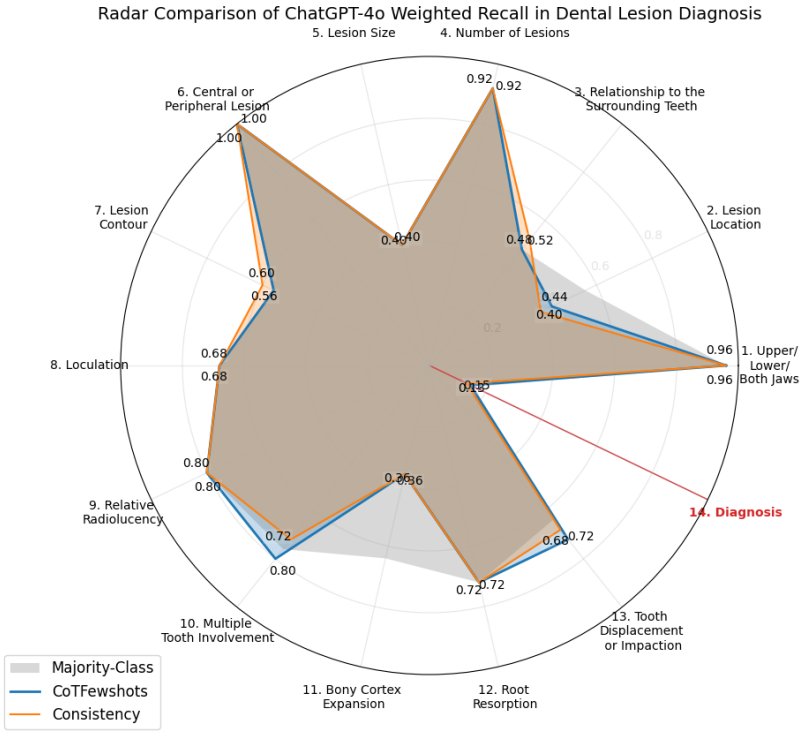

Figure 4.14: Per-question weighted recall for chain-of-thought with few-shot (blue) versus chain-of-thought with few-shot improved by self-consistency (orange). The overall performance of both approaches is very similar.

4.3.6 Self-Debate with Few-Shot

In this experiment, the thesis implements the *self-debate* strategy: GPT-4o Vision is prompted to pretend to be three different doctors, each offering an independent opinion on the panoramic image, and then to consolidate their viewpoints into a final consensus answer. Figure 4.15 shows the per-question weighted recall for standard chain-of-thought (CoT) with few-shot (blue) and CoT with few-shot augmented by self-debate (orange).

Key observations:

- Weighted recalls on Q3, Q5, Q7, Q11 and Q13 dropped enormously.
- For most of the other questions, the score also slightly dropped. Except for Q2, whose recall increased a bit.

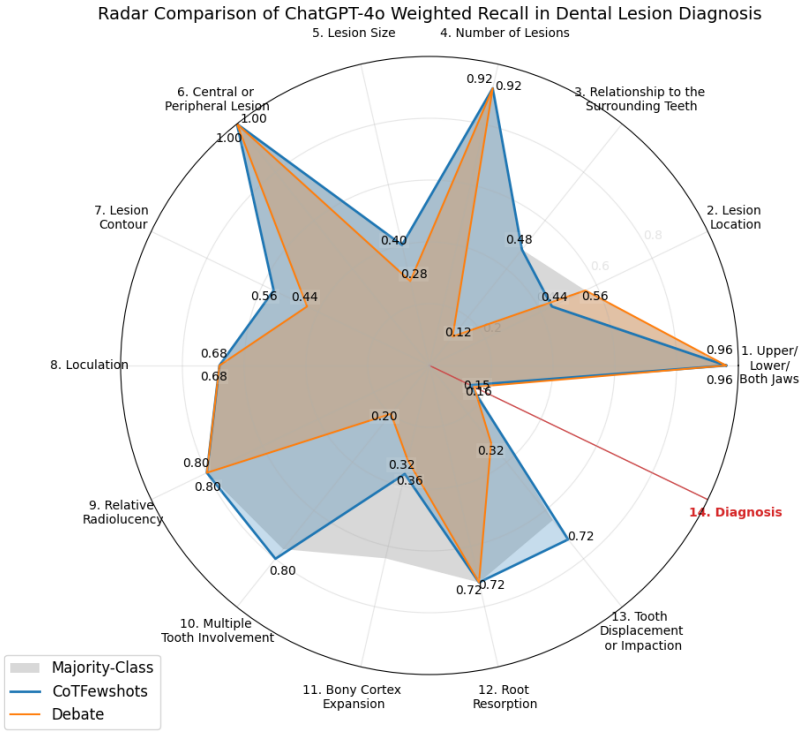

FIGURE 4.15: Per-question weighted recall for chain-of-thought with few-shot (blue) versus chain-of-thought with few-shot improved by self-debate (orange). Overall, self-debate reduces performance across most questions.

Analysis

To illustrate how self-debate shifts predictions, Figure 4.16 shows the answer distribution for Q10 ("Does the lesion include one or more teeth?"). The grey bars represent the ground-truth counts, while the blue bars show standard CoT few-shot and orange bars show CoT few-shot with self-debate. It can be found that the standard CoT few-shot closely matches the ground truth distribution. However, under self-debate, GPT-4o Vision predicted "No" only once and "Yes" 24 times, indicating that forcing diverse viewpoints actually confused the model on this question.

Conclusion

In summary, although self-debate was intended to broaden the model's reasoning, it introduced greater variance and led to a net decrease in weighted recall. The average weighted recall for self-debate is 0.534, compared to 0.641 for standard CoT

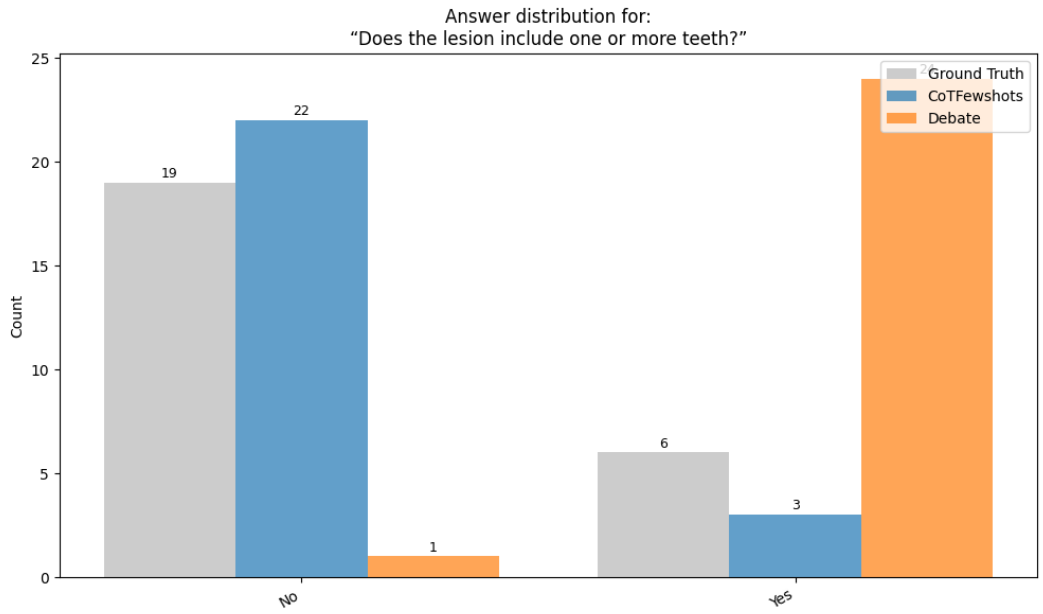

Figure 4.16: Answer distribution for Q10 (Multiple Tooth Involvement). Grey bars: ground truth. Blue bars: predictions with standard CoT + few-shot. Orange bars: CoT + few-shot with self-debate

few-shot. Because this represents an obvious performance drop, self-debate is not used to improve CoT. Therefore, self-debate is not used.

4.3.7 Self-Critique with Few-Shot

In this experiment, GPT-4o Vision was prompted to provide a self-critique immediately after each initial answer, with the intention of identifying and correcting any overlooked details before producing a final response. Figure 4.17 compares per-question weighted recall between standard chain-of-thought with few-shot (blue), and chain-of-thought with few-shot plus self-critique (orange).

Key observations:

- It can be observed that self-critique produces a modest improvement on Q2 (Lesion Location) and Q3 (Relationship to the Surrounding Teeth), whereas performance on Q10 (Multiple Tooth Involvement) and Q13 (Tooth Displacement or Impaction) is substantially reduced.
- Most other questions exhibit negligible change or a slight decline in weighted recall.

Conclusion

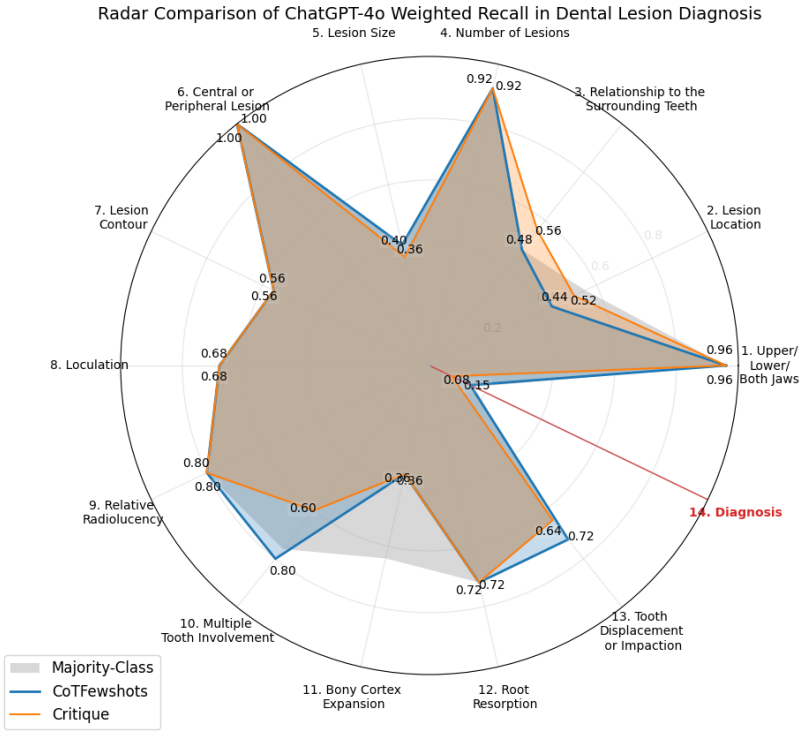

FIGURE 4.17: Comparison of weighted recall per question between standard CoT with few-shot and CoT with few-shot plus self-critique.

The effects of self-critique are inconsistent across the fourteen questions. Improvement depends heavily on GPT-4o Vision's internal confidence metrics, which are not externally visible. Because the overall weighted recall for CoT with few-shot plus self-critique (0.534) is markedly lower than that of standard CoT with few-shot (0.641), self-critique was not retained for subsequent experiments.

4.3.8 Summary

Table 4.2 summarizes each prompt-engineering variant, showing both the average weighted recall across all fourteen questions and the specific weighted recall on the diagnosis question (Q14). Since Chain-of-Thought (CoT) with Few-Shot achieves the highest overall score, it is chosen for all subsequent experiments.

Prompt Variant	Avg. Weighted Recall	Diagnosis (Q14) Recall
Zero-Shot	0.578	0.13
Zero-Shot (Free-Form)	0.558	0.13
Few-Shot	0.625	0.15
Few-Shot + Images	0.569	0.13
CoT + Few-Shot	0.641	0.15
ToT + Few-Shot	0.531	0.12
Self-Consistency + Few-Shot	0.635	0.13
Self-Debate + Few-Shot	0.534	0.16
Self-Critique + Few-Shot	0.534	0.08

Table 4.2: Average weighted recall across all fourteen diagnostic questions and the specific weighted recall on Q14 ("Diagnosis") for each prompt-engineering strategy.

4.4 Hyperparameter and Sequence Effect

4.4.1 Effect of Temperature

After major prompt-engineering techniques were explored, the effect of temperature configuration on result quality was studied. Figure 4.18 and Table 4.3 visualize the weighted recall across five temperature settings (0.00, 0.15, 0.30, 0.45, and 0.60). Chain-of-thought with few-shot prompting was used throughout this experiment due to its superior performance.

Key Observations
From the figure it can be seen that different temperatures slight alter the score, which can also because of the variance. However, the weighted recalls under temperature = 0.15 and 0.3 are reduced greatly compared with other results.

Temp	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14
0.00	0.96	0.44	0.52	0.92	0.40	1.00	0.52	0.68	0.80	0.80	0.36	0.72	0.72	0.15
0.15	0.96	0.48	0.04	0.92	0.40	1.00	0.56	0.68	0.80	0.68	0.36	0.72	0.60	0.23
0.30	0.96	0.40	0.48	0.92	0.40	1.00	0.60	0.68	0.80	0.64	0.36	0.72	0.60	0.19
0.45	0.92	0.44	0.48	0.92	0.40	1.00	0.56	0.68	0.80	0.64	0.32	0.72	0.48	0.19
0.60	0.92	0.40	0.04	0.92	0.36	1.00	0.48	0.68	0.80	0.68	0.45	0.72	0.56	0.04

Table 4.3: Weighted recall per question (Q1–Q14) at different temperature settings.

Analysis
From the table it can be seen that an increased temperature mainly decreases the weighted recall of many questions including Q1, Q3, Q10, Q13 and Q14. A Friedman test was performed on the weighted-recall scores for the fourteen questions across all five temperature settings. The resulting p-value was 0.0604, which exceeds the significance threshold of $\alpha = 0.05$. Hence, the null hypothesis (that temperature has

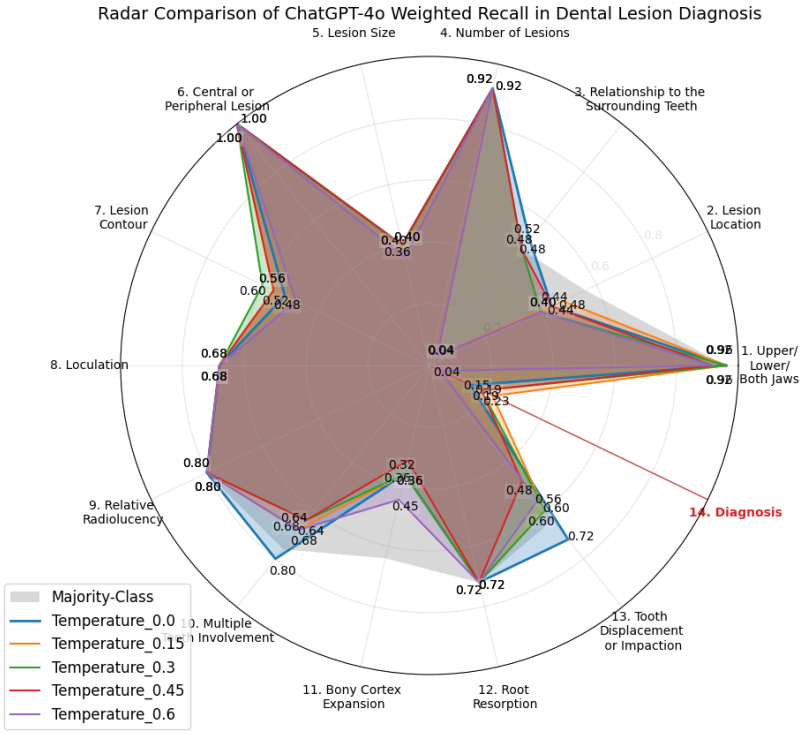

FIGURE 4.18: Weighted recall per question (Q1–Q14) under different temperature settings using CoT with few-shot prompting.

no effect on overall performance) cannot be rejected at the 5% level.

Conclusion
Based on the data from fourteen questions, temperature setting did not produce a statistically significant difference in weighted recall. However, given that $p = 0.0604$ is close to $\alpha = 0.05$, evaluation on a larger set of examples may yield a different outcome.

4.4.2 Effect of Question Order

The final experiment examined whether the order of the first thirteen questions affects response quality, with the fourteenth question (diagnosis) always posed last. Figure 4.19 and Table 4.4 summarize weighted-recall scores across five different random question-order permutations.

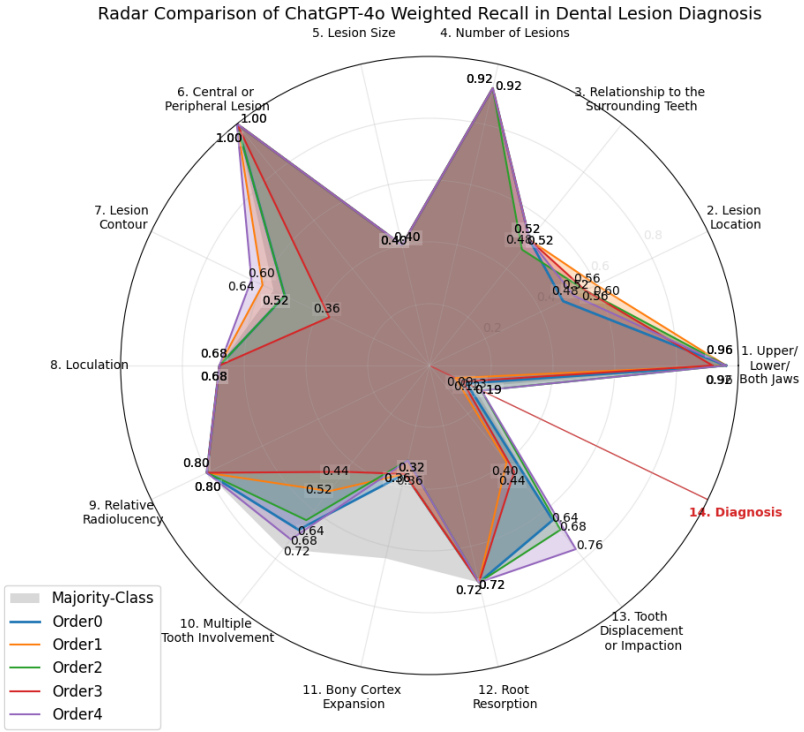

Figure 4.19: Weighted recall per question (Q1–Q14) under five question orders.

Key Observations

Different orders produce noticeable shifts in weighted recall distributions. In particular, Q2, Q7, Q10, and Q13 exhibit the largest variance across orders.

Order	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14
0	0.96	0.48	0.52	0.92	0.40	1.00	0.52	0.68	0.80	0.68	0.36	0.72	0.64	0.13
1	0.96	0.60	0.52	0.92	0.40	1.00	0.60	0.68	0.80	0.52	0.36	0.72	0.40	0.09
2	0.96	0.56	0.48	0.92	0.40	1.00	0.52	0.68	0.80	0.64	0.32	0.72	0.68	0.19
3	0.92	0.56	0.52	0.92	0.40	1.00	0.36	0.68	0.80	0.44	0.36	0.72	0.44	0.12
4	0.96	0.52	0.52	0.92	0.40	1.00	0.64	0.68	0.80	0.72	0.32	0.72	0.76	0.19

Table 4.4: Weighted recall per question (Q1–Q14) under five question-order permutations (values rounded to two decimal places).

Analysis

A Friedman test was applied to the fourteen questions across all five orders, yielding $p = 0.3553$. At the $\alpha = 0.05$ level, the null hypothesis (that question order has no effect on overall performance) cannot be rejected. However, variability appears concentrated in Q7, Q10, and Q13. A secondary Friedman test restricted to these three questions produced $p = 0.0641$ (freedom is set to 2 accordingly), which remains above 0.05 but is close enough to suggest a potential order effect if evaluated on a larger sample.

Conclusion
Question order does not significantly affect overall weighted recall at $\alpha = 0.05$. For individual questions, particularly Q7, Q10, and Q13, a larger dataset may reveal a statistically significant order effect. Therefore, this experiment reaches a conclusion that the order doesn't affect model performance, which can be because there are no dependencies among questions, or the previous wrong answers totally eliminate the dependencies.

4.5 Usage Scenarios

This section explores two practical ways in which GPT-4o Vision might be employed to enhance diagnostic accuracy when supplemental information is available.

4.5.1 Leave-One-Out

In this experiment, each question was answered with the correct labels provided for all other questions. For each of the fourteen questions, the model received ground-truth answers to the remaining thirteen and was then prompted to answer the withheld question using CoT with few-shot prompting (examples generated by the zero-shot model). Figure 4.20 illustrates the performance of this experiment.

Key Observations
- Eight of the fourteen questions show improved recall under the leave-one-out protocol, with particularly large gains for Q11 (Bony Cortex Expansion) and Q12 (Root Resorption).
- For Q3, Q5, Q8, Q10, Q11, Q12, Q13, and Q14, the weighted recall under leave-one-out exceeds the majority-class baseline, indicating that the model can reason more accurately when provided correct context for the other questions.
- Q14 (Diagnosis) improves from 0.15 to 0.28 in weighted recall.

Analysis
When the ground-truth labels for all other questions are supplied, GPT-4o Vision's accuracy on the remaining question increases substantially. This confirms that incorrect earlier answers in the standard chained setup were propagating errors to subsequent questions. Providing correct context allows the model to focus on a single question without compounding mistakes, revealing that a core bottleneck in the standard CoT chain arises from error accumulation rather than from inherent

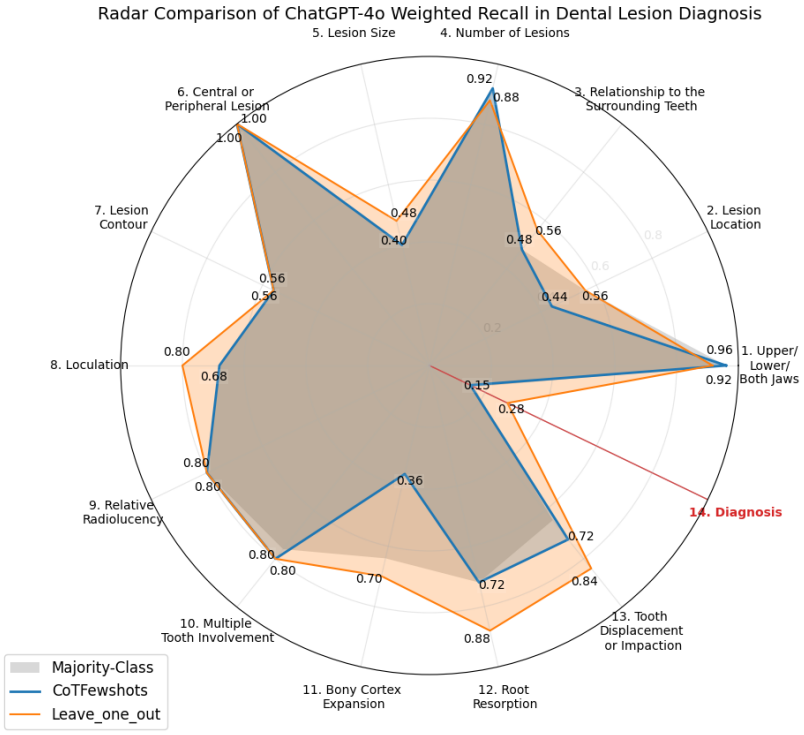

Figure 4.20: Weighted recall per question: standard CoT with few-shot (blue) versus leave-one-out (orange).

limitations of visual reasoning.

Conclusion
The leave-one-out results demonstrate that questions are not entirely independent: errors in initial answers degrade downstream performance. When correct information is given for the other questions, GPT-4o Vision's ability to answer individual questions improves notably, especially for Q11, Q12, and Q14. This suggests that preventing early misclassifications or providing reliable context significantly enhances overall diagnostic accuracy.

4.5.2 Correcting a Single Unknown Mistake

In this experiment, GPT-4o Vision (prompted via CoT with few-shot) receives correct answers for all but one question; the model is not told which answer is incorrect.

It must first identify the wrong label and then correct it, which mimics a real-world use case in which GPT-4o Vision assists in refining a student's diagnostic report by locating and amending a single erroneous response. Figure 4.21 shows two metrics per question: the accuracy of locating the incorrect answer (blue), and the accuracy of replacing it with the correct label (orange).

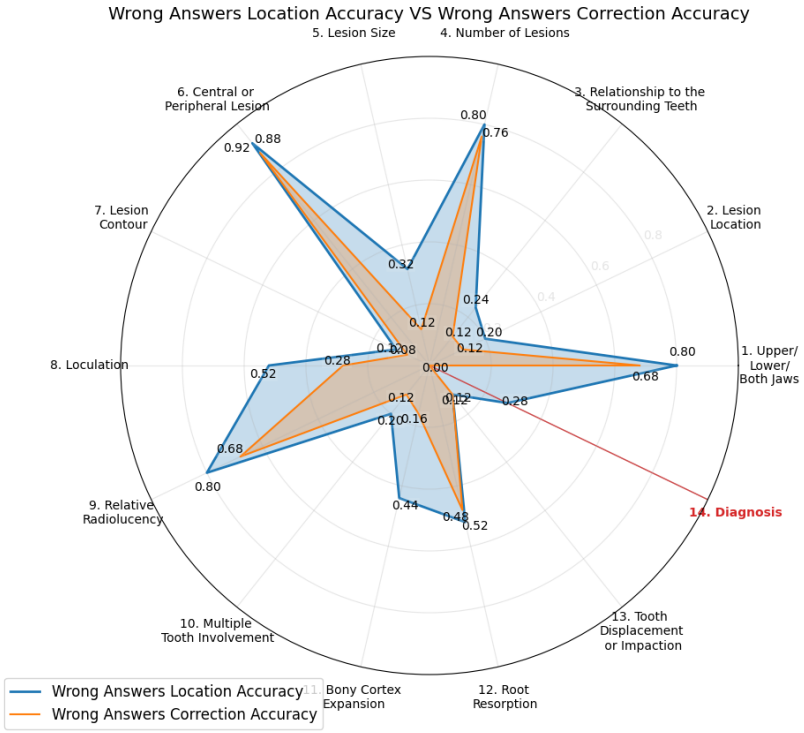

FIGURE 4.21: Per-question accuracy of (1) identifying which answer is wrong (blue) and (2) correcting that answer to the ground truth (orange).

Key Observations

- Q1, Q4, Q6, and Q9 exhibit high accuracy both in locating the wrong answer and correcting it.
- Q2, Q3, Q5, Q7, and Q10 show very low success rates for both identifying and correcting the error.
- For Q5, Q8, Q11, and Q14, the gap between locating the wrong answer and correcting it is substantial, indicating that even when the model guesses which label

is incorrect, it often fails to produce the correct replacement.

Analysis
Overall, correcting an unknown mistake is challenging for GPT-4o Vision. High performance on Q1, Q4, Q6, and Q9 likely reflects severe class imbalance (the majority class is trivially the correct label), making both locating and correcting the error easier by default. Conversely, low performance on Q2, Q3, Q5, Q7, and Q10 suggests that the model struggles to detect inconsistencies when visual cues are subtle or class distributions are more balanced. The large disparity between "locate" and "correct" success rates on certain questions (e.g., Q5, Q8, Q11, Q14) indicates that some predictions succeed by chance: the model may mark a label as wrong without having a reliable inference mechanism to propose the correct alternative.

Conclusion
GPT-4o Vision can correct certain easy-to-spot errors (Q1, Q4, Q6, Q9) when provided with all other ground-truth answers. However, this performance is largely driven by class imbalance rather than robust diagnostic reasoning. For most questions, the model fails to reliably identify and correct a single unknown mistake. This highlights limitations in GPT-4o Vision's ability to perform consistency checks and error correction under realistic diagnostic conditions. And given the poor performance for correcting 1 answer, it is unnecessary to test its ability to correct more wrong answers, which will only downgrade the performance.

4.6 Ablation Study

4.6.1 Answering a Single Question at a Time

Noting there are possible dependencies among questions, in this experiment, the chain-of-thought (CoT) with few-shot prompting framework was modified so that only a single question was presented to GPT-4o Vision at each prompt so that dependencies among questions are eliminated. This includes Q14 (diagnosis) on its own. Figure 4.22 displays the resulting weighted-recall scores for each of the fourteen questions when answered individually.

Key Observations

- Q10 (Multiple Tooth Involvement) and Q13 (Tooth Displacement or Impaction) exhibit a substantial drop in weighted recall—approximately 0.30—relative to standard CoT with few-shot. This aligns with prior findings that these questions are the least robust, with high variability across prompt variations.
- Q7 (Lesion Contour) shows an increase in weighted recall from 0.56 to 0.68. Most other questions remain effectively unchanged.
- Q14 (Diagnosis) remains highly challenging, with minimal improvement despite isolating the diagnostic question.

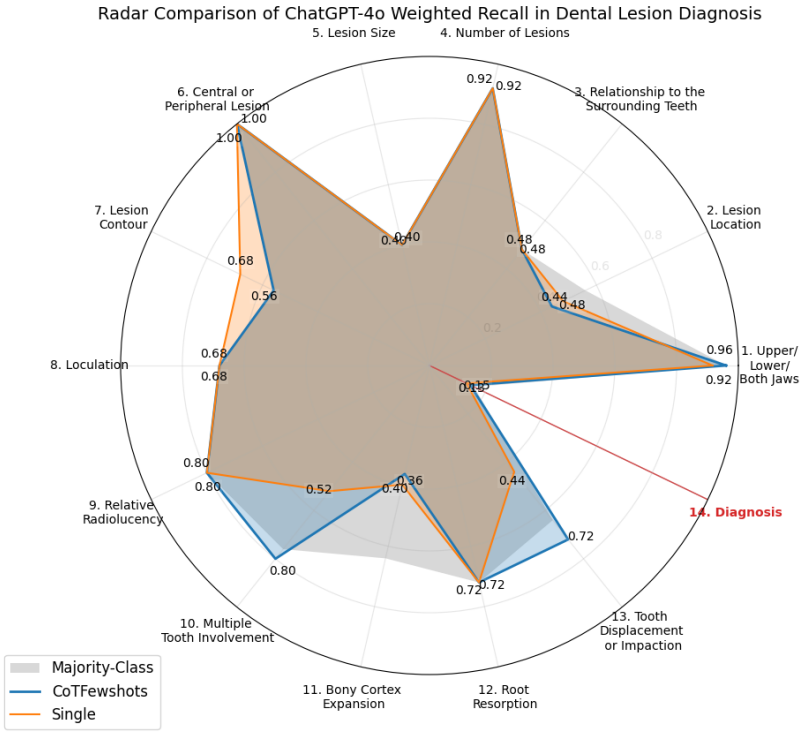

FIGURE 4.22: Weighted recall per question when GPT-4o Vision is prompted with CoT & few-shot, answering each question in isolation.

Analysis

The results indicate that, when dependencies are removed and each question is answered purely on visual information, most weighted-recall scores remain unchanged or decrease only slightly. This stability may occur because GPT-4o Vision is relying primarily on visual features or defaulting to the majority-class guess. Indeed, the fact that most scores align with the majority-class baseline suggests a high likelihood of majority-class prediction in the absence of contextual cues. However, the lack of decline in many questions could also imply that prior CoT answers were incorrect, thereby providing no useful context for subsequent questions, or that GPT-4o Vision processes each question independently—even when multiple questions are presented together—which seems unlikely. Furthermore, the substantial performance drop on Q10 and Q13 indicates that visual reasoning alone is not reliably sufficient for those tasks.

Conclusion
This ablation study shows a high probability that GPT-4o Vision is giving answers by guessing from the majority instead of getting clues from the images then answering accordingly.

4.6.2 Leave-One-Out without Images

In this experiment, the leave-one-out protocol was repeated, but no images were provided to the model. This setup evaluates whether GPT-4o Vision relies more on visual information or on textual inference when all but one answer are known. Figure 4.23 compares weighted recall per question for leave-one-out with images (blue) versus leave-one-out without images (orange).

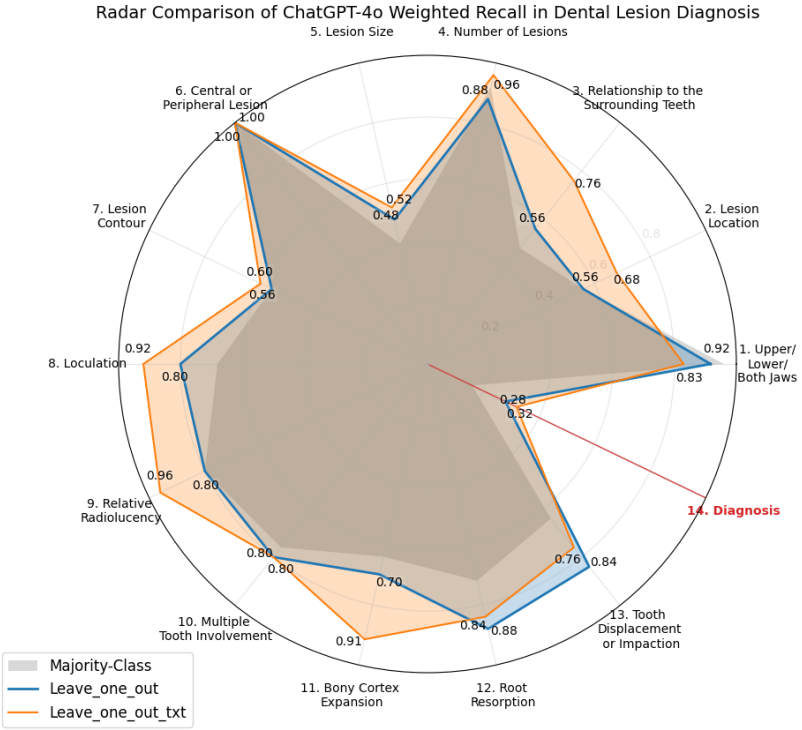

Figure 4.23: Weighted recall per question: leave-one-out with images (blue) versus without images (orange).

Key Observations
Surprisingly, omitting the image input yields higher overall performance. Notable

improvements appear in Q2, Q3, Q8, Q9, and Q11. Question Q14 (Diagnosis) remains challenging in both conditions.

Analysis
These results suggest that the visual cues provided by GPT-4o Vision's current vision model are often misleading, reducing accuracy on several questions. In the absence of images, the model appears to rely on textual context alone, which produces more accurate answers for most features. This outcome underscores that GPT-4o Vision's ability to interpret panoramic images is not reliably strong and would benefit from additional domain-specific fine-tuning.

Conclusion
Overall, GPT-4o Vision performs better on leave-one-out tasks when forced to rely solely on textual inference. The presence of image data, in its current form, often confuses the model rather than enhancing accuracy.

4.7 Summary

- **Baseline Performance**
  - Most classes exhibit severe imbalance, making a majority-class predictor deceptively strong.
  - In zero-shot mode, GPT-4o Vision often defaults to the majority label, yielding poor weighted recall.
  - With temperature=0 and top_p=0, variance across repeated zero-shot runs is negligible for all but one question, justifying one-shot evaluation under budget constraints.
  - Providing explicit answer choices (rather than allowing free-form responses) substantially improves performance.

- **Prompt Engineering Experiments**
  - Chain-of-Thought (CoT) yields the highest overall recall by guiding step-by-step reasoning.
  - Few-Shot examples boost performance by showing contrastive answer options, preventing the model from over-selecting the most frequent label.
  - Tree-of-Thought underperforms CoT when using GPT-4o Vision as its own evaluator; a domain-expert evaluator (e.g. a radiologist-curated rubric) would be needed to improve branching decisions.
  - Self-Critique and Self-Debate both degrade performance, indicating that GPT-4o Vision's own confidence estimates and alternate viewpoints introduce confusion rather than clarity.
  - Neither temperature nor question order has a statistically significant effect at $\alpha = 0.05$, though p-values near 0.05 suggest that larger sample sizes might reveal subtle effects. It indicates the dependencies among questions do not exist or is minimal.

- **Usage Scenarios**
  - In the leave-one-out setting, providing correct answers for all other questions dramatically improves weighted recall on the withheld question, confirming the dependency that earlier errors propagate downstream in a chained prompt.
  - When asked to detect and correct a single unknown mistake, GPT-4o Vision succeeds reliably only on highly imbalanced questions (e.g. Q1, Q4, Q6, Q9). Overall, the model cannot yet be trusted to identify and fix errors in a full answer set.

- **Ablation Study**
  - Answering each question in isolation shows that performance on many features remains at majority-class levels—indicating the model often guesses rather than uses visual clues. Questions whose recall drops significantly (e.g. Q10, Q13) reveal that GPT-4o Vision's raw image understanding is unreliable. Questions whose recall increases (Q7) indicates GPT-4o Vision may be good at this kind of question, and previous low score is due to the previous misleading results.
  - In leave-one-out without images, performance generally improves, demonstrating that the model's vision-based cues often mislead it; GPT-4o Vision appears to rely more on textual inference than on true image interpretation.

- **Diagnosis**
  Accurately predicting the final diagnosis (Q14) remains difficult in every experimental condition, even under "leave-one-out," where all other question answers are provided. This suggests that GPT-4o Vision, when limited to the thirteen feature questions and solely panoramic-image information, cannot reliably infer the correct lesion diagnosis. In practice, incorporating a vision encoder fine-tuned on dental radiology alongside supplementary patient data (e.g., clinical history, demographics) would likely be necessary to achieve meaningful diagnostic accuracy.

Chapter 5

Conclusion, Limitation and Future Work

In this thesis, GPT-4o Vision's performance was evaluated for annotating oral panoramic radiographs under various prompt-engineering strategies, contextual setups, and ablation studies. The goal was to assess its suitability as an image interpreter within retrieval-augmented generation (RAG) systems. The experiments yielded the following key findings:

1. **Overreliance on Textual Priors:**
   GPT-4o Vision frequently defaults to common answer choices learned from text rather than extracting visual features from the image itself.

2. **Benefit of Step-by-Step Reasoning:**
   Explicit chain-of-thought prompts significantly improve the model's ability to identify low-frequency or subtle radiographic patterns.

3. **Ineffectiveness of Multiple-Opinion Prompts:**
   Techniques that generate multiple reasoning branches, such as self-debate, self-critique, or tree-of-thought—consistently degraded performance instead of enhancing it.

4. **Unreliable Image Annotation:**
   In its current form, GPT-4o Vision's annotations of panoramic images are highly inconsistent and often produce misleading conclusions when combined with textual context.

5. **Question Dependencies and Error Propagation:**
   When all questions are presented together, earlier incorrect answers can negatively influence subsequent predictions, introducing a harmful dependency among questions.

The following sections analyze these findings in detail, discuss the limitations of this study, and outline directions for future work.

5.1 Conclusions Analysis

5.1.1 Overreliance on Textual Priors

Many questions in this dataset exhibit extreme class imbalance (e.g., "Which jaw is affected?" or "Does the lesion cause root resorption?"). As a result, a predictor that always selects the majority label achieves a deceptively high weighted recall. Under zero-shot prompting (Experiment 4.2.2), GPT-4o Vision frequently defaulted to the dominant class without performing any meaningful image analysis. Its per-question recall closely tracked the underlying class frequencies. Even for questions with only 40–50 % support for the majority label (such as "Lesion location" and "Lesion size"), zero-shot recall remained well below that baseline, suggesting uncertainty rather than genuine visual reasoning.

This pattern persisted across nearly all experimental settings. In the "Answering a Single Question at a Time" ablation (Experiment 4.6.1), where contextual cues from other answers were removed, most questions' recall scores still aligned with the majority-class baseline. In other words, even when forced to rely solely on the image, GPT-4o Vision often reverted to its text-based priors.

In conclusion, **these findings indicate that GPT-4o Vision typically guesses the most frequent answer from its pretrained textual knowledge instead of extracting diagnostic features from the panoramic radiograph.**

5.1.2 Benefit of Step-by-Step Reasoning

When prompted to "think aloud" via a chain-of-thought (CoT) prompt, GPT-4o Vision's average weighted recall increases from 0.625 to 0.641 (Experiment 4.3.3). **One possible explanation is that encouraging the model to articulate intermediate reasoning appears to discourage immediate majority-class guessing and instead promotes deeper use of visual and contextual clues.** As a result, GPT-4o Vision can more effectively integrate prior answers into its final decision, yielding consistent gains across multiple diagnostic questions.

5.1.3 Ineffectiveness of Multiple-Opinion Prompts

After establishing CoT+Few-Shot as the best prompt, additional intelligence-enhancing strategies were tested, including multiple chains of thought (Tree-of-Thoughts, ToT), self-consistency voting, self-debate, and self-critique (Experiment 4.3.4 - 4.3.7). Each strategy reduced performance. In the ToT setup, the model's evaluator proves unreliable. Its scoring criteria are unclear, leading it to favor superficially plausible branches that ultimately guide the search toward incorrect decisions. This reduced recall from 0.641 (CoT+Few-Shot) to 0.531. Self-consistency, which runs CoT five times and takes the majority vote, yielded a marginal change (0.635). Self-debate and self-critique reduced performance to the low 0.53 range. **In all cases, these**

additional reasoning steps introduced confusion by amplifying linguistic priors not grounded in visual understanding.

5.1.4 Unreliable Image Annotation

Across multiple ablations, including visual input often degraded GPT-4o Vision's performance. First, augmenting few-shot examples with their corresponding images led to a drop in weighted recall (0.625 → 0.569), showing that adding radiographic context confused the model rather than clarifying it (Experiment 4.3.1). Second, when each question was answered in isolation, which forces the model to rely solely on vision rather than contextual cues, performance on several questions fell sharply (e.g., Q10 and Q13 dropped by 0.30) (Experiment 4.6.1). Finally, in the leave-one-out ablation, omitting the image entirely improved weighted recall on most questions compared to using both image and context (Experiment 4.6.2). Together, these results demonstrate that, **without domain-specific fine-tuning, GPT-4o Vision's image encoder misreads subtle panoramic features and can be misled by its visual input.**

5.1.5 Question Dependencies and Error Propagation

The fact that randomizing question order did not significantly change weighted recall (Experiment 4.4.2) suggests either that GPT-4o Vision does not inherently rely on question sequencing or that any latent dependencies are effectively broken when earlier answers are incorrect. In contrast, the leave-one-out experiments, where ground-truth answers for all but one question were provided, produced marked improvements in weighted recall on the withheld item (Experiment 4.5.1). This indicates that, when preceding labels are accurate, the model can use those correct contexts to enhance its inference. However, once the model's own earlier answers deviate from the ground truth, those errors propagate and disrupt any beneficial dependencies. **In other words, question dependencies exist and can improve performance when prior labels are correct, but they become harmful whenever the model's initial predictions are wrong.**

5.1.6 Final Conclusion

GPT-4o Vision consistently prioritizes textual context over visual information, and its interpretations of dental panoramic images often introduce errors rather than correcting them. Consequently, in its current form, GPT-4o Vision is not suitable for direct integration into multimodal RAG pipelines; reliable performance will require domain-specific fine-tuning and expert-guided adjustments.

5.2 Limitations and Future Work

This thesis has several limitations that future research could address to provide a more comprehensive and nuanced evaluation of vision-enabled GPT models in

medical imaging contexts.

5.2.1 Small and Imbalanced Dataset

The evaluation relies on only 25 cases, repeatedly used across prompt variants. While this allowed exhaustive exploration of prompt-engineering choices, it does not capture the diversity of real-world panoramic radiographs. The dataset's severe class imbalance may bias results, and it is unclear if its label distribution reflects clinical reality. Moreover, the Friedman test p-values for temperature and question-order effects are very close to the 5% significance threshold; a larger sample might yield different conclusions.

Future Work:
Assemble a larger, more balanced dataset (e.g., 100–200 cases) to confirm whether temperature or question order truly has no effect, and to produce more statistically robust evaluations of vision-GPT performance.

5.2.2 Budget and Time Constraints

This thesis had no dedicated budget, and each API call with GPT-4o Vision incurs significant cost. Although free models like LLaVA and Qwen are available, their performance, stability, and latency were inferior. Consequently, most experiments were run only once; the low variance observed in zero-shot tests partly justifies this, but multiple runs per setting would yield more reliable mean and variance estimates. Time limitations also prevented exploring additional variables (e.g., varying the number of few-shot examples or self-debate participants).

Future Work:
Allocate sufficient funds or compute credits to repeat each experiment multiple times, thereby estimating both average performance and variability. Systematically vary prompt parameters, such as few-shot example count, example content, or the number of "experts" in self-debate, to understand their individual contributions to performance stability.

5.2.3 Lack of Professional Clinical Input

Throughout this work, example cases and evaluation criteria were designed without direct guidance from oral radiologists. As a result, few-shot examples generated via zero-shot bootstrapping may reinforce model biases, and the Tree-of-Thought evaluator (GPT-4o Vision itself) proved unreliable. Expert input could improve both the prompts and the evaluation metrics.

Future Work:
Collaborate with dental radiologists to curate high-quality few-shot examples rather than relying on model-generated outputs. Design a rule-based or clinician-verified evaluator for Tree-of-Thought prompting, ensuring that candidate reasoning paths are scored according to domain-specific criteria rather than generic language fluency.

5.2.4 Superficial Result Interpretation

This thesis emphasizes quantitative performance metrics and high-level analyses (e.g., "overreliance on textual priors"), but it does not deeply investigate how GPT models' internal mechanisms produce these outcomes. A richer theoretical or probing analysis could reveal exactly which internal representations or attention patterns cause failure modes in panoramic image interpretation.
Future Work:
Conduct targeted ablations or representation-level analyses (e.g., attention maps, embedding distributions) in architecture level to identify precisely why GPT-4o Vision misinterprets certain radiographic features. Formulate hypotheses, such as "the model's vision encoder fails to localize cortical borders", and test them with controlled experiments.

5.3 Future Directions: Toward a Full Multimodal RAG System

Based on the foregoing findings, the following research directions are recommended to build a robust multimodal retrieval-augmented generation (RAG) pipeline for dental radiology:

5.3.1 Clinical Fine-Tuning of the Vision Encoder

Fine-tuning (or LoRA-adapting) a ViT-style vision backbone on a larger corpus of annotated panoramic radiographs could substantially improve feature extraction. But this requires a large amount of images with labels for each dental pathology class (e.g., cortical expansion, root resorption, tooth impaction).

5.3.2 Incorporating Multimodal Inputs

Although GPT-4o Vision is capable of ingesting images and text, this work focused on image-only annotation. Incorporating patient metadata, such as age, symptoms, or laboratory values, would help disambiguate visually similar findings. Allowing the model to process structured clinical data (eg. medical record) or multilingual radiology reports could further ground its reasoning. Furthermore, it is worth trying masking panoramic images with tooth segmentation at first to help vision-GPT understand the images.

5.4 Closing Remarks

In summary, this thesis has demonstrated that, although GPT-4o Vision shows promise, its heavy dependence on textual priors and inconsistent interpretation of panoramic images precludes its direct integration into multimodal RAG pipelines for dental radiology. Extensive prompt engineering and ablation studies revealed both strengths (e.g., improved performance with step-by-step reasoning) and fundamental

weaknesses (e.g., error propagation and misleading visual cues). To overcome these limitations, future work must include domain-specific fine-tuning, expert-curated examples, and richer multimodal context. While considerable challenges remain, this research establishes a clear foundation for developing more reliable, clinically robust vision-GPT systems in oral healthcare.

Appendices

Appendix A

Fine-Tuning ViT on 25 Panoramic Radiographs

To assess whether a standard Vision Transformer (ViT) could be adapted to the oral radiology domain using only 25 annotated cases, the thesis attempted to fine-tune a pre-trained ViT model end-to-end. Below is a concise description of the procedure, hyperparameters, and observed outcomes, demonstrating why this approach proved infeasible with so few examples.

A.1 Dataset Preparation

The 25 cases were split into 20 images for training and 5 for validation. Each panoramic radiograph was resized to 224×224 pixels (the ViT-Base input resolution) and normalized according to the same mean and standard deviation used during ViT's ImageNet pre-training. The goal here was to predict a single diagnostic label (e.g. Ameloblastoma) as a proof of concept, a linear classification head is attached on top of the ViT's [CLS] token.

Model Architecture and Initialization

A ViT-Base-224 checkpoint, pretrained on ImageNet-21k, was employed. Two fine-tuning strategies were attempted:

1. **Full Fine-Tuning:** All transformer parameters (self-attention layers, MLPs, embedding layers) were left unfrozen and updated. A new linear layer (output size = 9) was randomly initialized and appended to the existing [CLS] projection, resulting in approximately 86 million trainable parameters.
2. **Head-Only Fine-Tuning:** After observing rapid overfitting when updating all 86 million parameters, the ViT encoder was frozen and only the newly added 9-class linear head was trained. This reduced the number of trainable parameters to just the classification layer, while the ViT backbone remained fixed.

A.2 Training Configuration

A lightweight training script was used with the following hyperparameters, chosen to mirror typical ViT-fine-tuning recipes:

- **Optimizer:** AdamW with weight decay = 0.05
- **Learning Rate (LR):** 1×10^{-4} (no scheduler)
- **Batch Size:** 4 images per GPU
- **Epochs:** 50
- **Data Augmentation:** Random horizontal flip and slight color jitter (brightness/contrast $\pm 10\%$)
- **Loss Function:** Cross-entropy
- **Validation:** Compute accuracy and loss on the held-out five images after each epoch

A.3 Observation

Despite trying multiple learning rates ($1 \times 10^{-5}, 5 \times 10^{-5}, 2 \times 10^{-4}$) and reducing weight decay to as low as 0.01, and using various learning rate schedulers, the rapid overfitting is always observed in both experiments. After only 3-5 epochs, the training accuracy rises to 100% while the validation accuracy always keeps at 0.

A.4 Conclusion

The rapid overfitting and failure in generalization indicate that fine-tuning a full ViT on only 25 panoramic radiographs did not yield a model capable of generalizing beyond the training set, regardless of whether it was achieved with a new classification head or by updating the entire model. Therefore, to fine-tune the ViT backbone, a larger dataset is necessary.

Appendix B

Prompts

In this appendix, all prompts used in prompt engineering experiments for GPT-4o Vision are listed. Prompts used for Grok and Claude are almost the same, except minor change in roles and the name of image information.

B.1 Zero-shot

Listing B.1: Zero-shot prompt template sent to the OpenAI API

```
messages = [
    {
        "role": "system",
        "content": (
            "You are an oral radiology expert assistant. "
            "Analyze the panoramic image thoroughly. "
            "Then, for each numbered question, give a concise final answer
            drawn from the listed options. "
            "Do not say "I can't analyze images.""
            "This is for exploratory/educational use only."
        )
    },
    {
        "role": "user",
        "content": [
            {
                "type": "text",
                "text": (
                    """
                    Below is the panoramic image that you are going to help
                    annotate.
                    """
                )
            },
            {
                "type": "image_url",
                "image_url": {
```

```
                    "url": "<BASE64_IMAGE>"
                }
            },
            {
                "type": "text",
                "text": (
                    "Answer each question by number, selecting from the
                    provided choices."
                    "For the final question, provide your own
                    free-form response."
                    f"{questions}\n\n"
                    "Finally, list each question number followed by
                    your answer."
                )
            }
        ]
    }
]
```

B.2 Zero-shot-free

Listing B.2: Zero-shot-free prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                "You are an oral radiology expert assistant. "
                "Analyze the oral panoramic image thoroughly and provide your
                   internal reasoning in free-form. "
                "Then, for each numbered question, give a concise final answer
                   drawn from the listed options. "
                "Do not say "I can't analyze images.""
                "This is for exploratory/educational use only."
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": "Here is the oral panoramic image to annotate:"
                },
                {
                    "type": "image_url",
                    "image_url": {
                        "url": f"{base64_image}"
                    }
                },
```

```
                {
                    "type": "text",
                    "text": (
                        "Answer each question by number."
                        "For the first 13 questions, there are also some
                            answers for reference."
                        "You can generate your responses in free-form."
                        f"{questions}\n\n"
                        "Finally, list each question number followed by your
                            answer."
                    )
                }
            ]
        }
    ]
```

B.3 Few-shot

Listing B.3: Few-shot prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                "You are an oral radiology expert assistant. "
                "Analyze the oral panoramic image thoroughly. "
                "Then, for each numbered question, give a concise final answer
                    drawn from the listed options. "
                "Do not say "I can't analyze images.""
                "This is for exploratory/educational use only."
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below are two sets of example answers with the
                            corresponding questions for you to generate more
                            accurate responses.
                        """
                    )
                },
            ]
        },
        {
            "role": "assistant",
```

```
            "content": (
                f"""
                {examples}
                """
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below is the image that you are going to help annotate.
                        """
                    )
                },
                {
                    "type": "image_url",
                    "image_url": {
                        "url": f"{base64_image}"
                    }
                },
                {
                    "type": "text",
                    "text": (
                        f"""
                        "Answer each question by number, selecting from the
                           provided choices."
                        "For the final question, provide your own free-form
                           response."
                        f"{questions}\n\n"
                        "Finally, list each question number followed by your
                           answer."
                        """
                    )
                }
            ]
        }
    ]
```

B.4 Few-shot-Img

Listing B.4: Few-shot-Img prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
```

```
        "You are an oral radiology expert assistant. "
        "Analyze the oral panoramic image thoroughly. "
        "Then, for each numbered question, give a concise final answer
           drawn from the listed options. "
        "Do not say "I can't analyze images.""
    )
},
{
    "role": "user",
    "content": [
        {
            "type": "text",
            "text": (
                """
                Below are two sets of example answers with the
                   corresponding questions for you to generate more
                   accurate responses.
                """
            )
        },
    ]
},
{
    "role": "user",
    "content": [
        {
            "type": "image_url",
            "image_url": {
                "url": f"{img64_exp1}"
            }
        }
    ]
},
{
    "role": "assistant",
    "content": (
        f"""
        {example1}
        """
    )
},
{
    "role": "user",
    "content": [
        {
            "type": "image_url",
            "image_url": {
                "url": f"{img64_exp2}"
            }
        }
```

```
            ]
        },
        {
            "role": "assistant",
            "content": (
                f"""
                {example2}
                """
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below is the image that you are going to help annotate.
                        """
                    )
                },
                {
                    "type": "image_url",
                    "image_url": {
                        "url": f"{base64_image}"
                    }
                },
                {
                    "type": "text",
                    "text": (
                        f"""
                        "Answer each question by number, selecting from the
                           provided choices."
                        "For the final question, provide your own free-form
                           response."
                        f"{questions}\n\n"
                        "Finally, list each question number followed by your
                           answer."
                        """
                    )
                }
            ]
        }
    ]
```

B.5 CoT with Few-shot

Listing B.5: CoT with Few-shot prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                "You are an expert assistant in annotating panoramic image of
                   oral lesions. "
                "Use a chain-of-thought approach: think step by step before
                   giving each final answer. "
                "Do not refuse by saying 'I can't analyze images.' "
                "This is for exploratory/educational use only."
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below are two sets of example answers with the
                           corresponding questions for you to generate more
                           accurate responses.
                        """
                    )
                },
            ]
        },
        {
            "role": "assistant",
            "content": (
                f"""
                {examples}
                """
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below is the image that you are going to help annotate.
                        """
                    )
                },
                {
                    "type": "image_url",
                    "image_url": {
```

```
                        "url": f"{base64_image}"
                    }
                },
                {
                    "type": "text",
                    "text": (
                        "Let's think step by step.\n\n"
                        "For each of the following questions, provide your
                            internal reasoning followed by "
                        "a final answer (choose from the provided options for
                            Q1-Q13, free-form for Q14):\n\n"
                        f"{questions}\n\n"
                        "At the end, list each question number and your answer
                            ."
                    )
                }
            ]
        }
    ]
```

B.6 ToT Evaluator

Listing B.6: ToT Evaluator prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                "You are an expert assistant in annotating panoramic image of
                    oral lesions. "
                "You are going to judge how confidently correct an answer to a
                    question is regarding to the image."
                "Do not refuse by saying 'I can't analyze images.' "
                "This is for exploratory/educational use only."
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below is the image which the question and answer is
                            about.
                        """
                    )
                },
                {
```

```
                    "type": "image_url",
                    "image_url": {
                        "url": f"{image_b64}"
                    }
                }
            ]
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        f"""
                        Below is the question and the associated reasoning and
                            answer.
                        {question}
                        {branch_text}
                        On a scale from 0.0 (no confidence) to 1.0 (maximum
                            confidence), how confident are you that you think
                            it is correct?
                        Please reply with a single floating-point number.
                        """
                    )
                },
            ]
        }
    ]
```

B.7 ToT Candidates Generation with Few-shot

Listing B.7: ToT Candidates Generation with Few-shot prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                "You are an expert assistant in annotating panoramic image of
                    oral lesions. "
                "Use a chain-of-thought approach: think step by step before
                    giving each final answer. "
                "Do not refuse by saying 'I can't analyze images.' "
                "This is for exploratory/educational use only."
            )
        },
        {
            "role": "user",
```

```
    "content": [
        {
            "type": "text",
            "text": (
                """
                Below are two sets of example answers with the
                   corresponding questions for you to generate more
                   accurate responses.
                """
            )
        },
    ]
},
{
    "role": "assistant",
    "content": (
        f"""
        {examples}
        """
    )
},
{
    "role": "user",
    "content": [
        {
            "type": "text",
            "text": (
                """
                Below is the image that you are going to help annotate.
                """
            )
        },
        {
            "type": "image_url",
            "image_url": {
                "url": f"{image_b64}"
            }
        }
    ]
},
{
    "role": "assistant",
    "content": [
        {
            "type": "text",
            "text": (
                f"""
                The following are the already answered questions with
                   the associated answers.
                {previous_answers}
```

```
                    """
                )
            }
        ]
    },
    {
        "role": "user",
        "content": [
            {
                "type": "text",
                "text": (
                    "Let's think step by step.\n\n"
                    "For the following question, provide your internal
                       reasoning followed by "
                    "a final answer (choose from the provided options for
                       Q1-Q13, free-form for Q14 which is about making a
                       diagnosis):\n\n"
                    f"{q_idx+1}. {next_q}\n"
                    "Just answer this question."
                )
            }
        ]
    }
]
```

B.8 Self-debate with Few-shot

Listing B.8: Self-debate with Few-shot prompt template sent to the OpenAI API

```
messages = [
    {
        "role": "system",
        "content": (
            """
            You are an expert assistant in understanding and annotating
               panoramic image of oral lesions.
            Your task is to analyze the provided image and answer the user'
               s questions.
            Do not refuse to answer by saying 'I can't analyze images'
                  instead  , adapt and provide the best possible response.

            IMPORTANT: "You are three radiology experts (Dr. A, Dr. B, and
               Dr. C) debating each question "
            "about the oral panoramic image.  \n"
            "1) Each doctor gives a short rationale.  \n"
            "2) They may disagree.  \n"
            "3) At the end, produce a single, unified answer for each
               question."
```

```
        Do not refuse to answer by stating "I can't analyze images."
           Instead, adapt and provide the best possible response based
            on the image data.
        Please provide your best-guess interpretation of the oral
           panoramic image along with your chain-of-thought reasoning.
            I understand that your analysis may not be clinically
           accurate and this is solely for educational or exploratory
           purposes, not for actual diagnosis.
        """
    )
},
{
    "role": "user",
    "content": [
        {
            "type": "text",
            "text": (
                """
                Below are two sets of example answers for you to
                   generate more accurate responses.
                """
            )
        }
    ]
},
{
    "role": "assistant",
    "content": (
        f"""
        {examples}
        """
    )
},
{
    "role": "user",
        "content": [
            {
                "type": "text",
                "text": (
                    """
                    Below is the image that you are going to help
                       annotate.
                    """
                )
            },
            {
                "type": "image_url",
                "image_url": {
                    "url": f"{base64_image}"
                }
```

```
                },
                {
                    "type": "text",
                    "text": (
                        f"""
                        Please answer the following questions, the possible
                            answers are provided as well:
                        {questions}
                        """
                    )
                }
            ]
        }
    ]
```

B.9 Self-critique with Few-shot

Listing B.9: Self-critique with Few-shot prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                """
                You are an expert assistant in understanding and annotating
                    oral panoramic imageof oral lesions.
                Your task is to analyze the provided image and answer the user'
                    s questions.
                Do not refuse to answer by saying 'I can't analyze images'-
                    instead, adapt and provide the best possible response.

                IMPORTANT: "For each question, you will:  \n"
                "  1) Give your best initial answer, using chain-of-thought
                    optional.  \n"
                "  2) Critically review your own answer, pointing out any
                    weaknesses, ambiguities, or alternative interpretations.  \
                    n"
                "  3) Provide a final, revised answer drawing on that self-
                    critique."
                Do not refuse to answer by stating "I can't analyze images."
                    Instead, adapt and provide the best possible response based
                     on the image data.
                Please provide your best-guess interpretation of the oral
                    panoramic image along with your chain-of-thought reasoning.
                     I understand that your analysis may not be clinically
                    accurate and this is solely for educational or exploratory
                    purposes, not for actual diagnosis.
                """
```

```
    )
},
{
    "role": "user",
    "content": [
        {
            "type": "text",
            "text": (
                """
                Below are two sets of example answers for you to
                    generate more accurate responses.
                """
            )
        }
    ]
},
{
    "role": "assistant",
    "content": (
        f"""
        {examples}
        """
    )
},
{
    "role": "user",
        "content": [
            {
                "type": "text",
                "text": (
                    """
                    Below is the image that you are going to help
                        annotate.
                    """
                )
            },
            {
                "type": "image_url",
                "image_url": {
                    "url": f"{base64_image}"
                }
            },
            {
                "type": "text",
                "text": (
                    f"""
                    Please answer the following questions, the possible
                        answers are provided as well:
                    {questions}
```

```
                                For each question, follow the 3-step process
                               above.  \n
                                At the very end, list "Final Answers:" and give
                                Q1-Q14 with your revised choice.
                            """
                        )
                    }
                ]
        }
    ]
```

B.10 Leave One Out

Listing B.10: Leave one out prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                "You are an expert assistant in annotating panoramic image of
                   oral lesions. "
                "Use a chain-of-thought approach: think step by step before
                   giving each final answer. "
                "Do not refuse by saying 'I can't analyze images.' "
                "This is for exploratory/educational use only."
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below are two sets of example answers with the
                           corresponding questions for you to generate more
                           accurate responses.
                        """
                    )
                },
            ]
        },
        {
            "role": "assistant",
            "content": (
                f"""
                {examples}
                """
```

```
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below is the image that you are going to help annotate.
                        """
                    )
                },
                {
                    "type": "image_url",
                    "image_url": {
                        "url": f"{base64_image}"
                    }
                },
                {
                    "type": "text",
                    "text": (
                       f"""
                        We also learned the following information from a
                            professional doctor:
                        {info}
                        """
                    )

                },
                {
                    "type": "text",
                    "text": (
                       f"""
                        Please give the answer for the following question
                            according to the image and the information from the
                             doctor, possible answers are provided as well:
                        {query}
                        """
                    )
                }
            ]
        }
    ]
```

B.11 Leave One Out (Txt only for ablation)

Listing B.11: Leave One Out (Txt only for ablation) prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                """
                "You are an expert assistant in annotating oral lesions. "
                "Use a chain-of-thought approach: think step by step before
                   giving each final answer. "
                "Do not refuse by saying 'I can't analyze.' "
                "This is for exploratory/educational use only."
                """
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
                        Below are two sets of example answers with the
                           corresponding questions for you to generate more
                           accurate responses.
                        """
                    )
                },
            ]
        },
        {
            "role": "assistant",
            "content": (
                f"""
                {examples}
                """
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                       f"""
                        Now you need to answer the user's question.
                        We also learned the following information from a
                           professional doctor:
                        {info}
                        """
```

```
                    )

                },
                {
                    "type": "text",
                    "text": (
                       f"""
                        Please give the answer for the following question:
                        {query}
                        """
                    )
                }
            ]
        }
    ]
```

B.12 Correct One Unknown Mistake

Listing B.12: Correct One Unknown Mistake prompt template sent to the OpenAI API

```
messages = [
        {
            "role": "system",
            "content": (
                """
                You are an expert assistant in understanding and annotating
                   panoramic image of oral lesions.
                Your task is to correct the mistake in an annotation of the
                   image given by a doctor student.
                There is onle one mistake in the given annotation, but you don'
                   t know which one is incorrect.

                Use a chain-of-thought approach: think step by step before
                   giving each final answer.
                Do not refuse by saying 'I can't analyze images.'
                This is for exploratory/educational use only.
                """
            )
        },
        {
            "role": "user",
            "content": [
                {
                    "type": "text",
                    "text": (
                        """
```

```
                Below are two sets of example answers with the
                    corresponding questions for you to generate more
                    accurate responses.
                """
            )
        },
    ]
},
{
    "role": "assistant",
    "content": (
        f"""
        {examples}
        """
    )
},
{
    "role": "user",
    "content": [
        {
            "type": "text",
            "text": (
                """
                Below is the image that you are going to help annotate.
                """
            )
        },
        {
            "type": "image_url",
            "image_url": {
                "url": f"{base64_image}"
            }
        },
        {
            "type": "text",
            "text": (
               f"""
                The following annotation is from a doctor student,
                    which includes one mistake:
                {info}
                """
            )

        },
        {
            "type": "text",
            "text": (
               f"""
                Please correct the mistake by directly generating the
                    correct answer together with the associated
```

```
                        question whose answer you think is wrong.
                    """
                )
            }
        ]
    }
]
```

B.13 Answer Extractor

Listing B.13: Prompt template sent to the OpenAI API for the answer extractor

```
messages = [
    {
        "role": "system",
        "content": (
            f"""
            You are an expert in information extraction.
            Your task is to generate a json object, with keys the questions
                and values the extracted corresponding answers.
            Output only valid JSON with no Markdown formatting or triple
               backticks.
            Below are the 14 questions and the allowed answers you are
               going to consider, except the final quesiton which is about
                diagnosis.
            Pick exactly one answer for the first 13 questions.
            For the final (14th) question, provide your own free-form
               response.
            {questions}
            """
        )
    },
    {
        "role": "system",
        "content": (
            """
            Use the following structure for each item in the json file:
            - <Nr>. <Question>: <Answer>

            For example:
            - 12. Does the lesion result in root resorption?: No",

            There are 14 questions, do not miss anyone.
            """
        )
    },
    {
        "role": "user",
        "content": [
```

```
                {
                    "type": "text",
                    "text": (
                        f"""
                        Below is the text you are going to use to extract
                            answers and store as a json file.
                        {query}
                        """
                    )
                }
            ]
        }
    ]
```

Bibliography

[1] Dandan Wang and Shiqing Zhang. Large language models in medical and healthcare fields: applications, advances, and challenges. *Artificial Intelligence Review*, 57(11):299, 2024.

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[3] Chiu-Liang Liu, Chien-Ta Ho, and Tzu-Chi Wu. Custom gpts enhancing performance and evidence compared with gpt-3.5, gpt-4, and gpt-4o? a study on the emergency medicine specialist examination. In *Healthcare*, volume 12, page 1726. MDPI, 2024.

[4] DeepScribe AI. Deepscribe outperforms gpt-4 by 59% on ai medical scribing: A benchmark study. https://www.deepscribe.ai/resources/deepscribe-outperforms-gpt-4-by-59-percent-on-ai-medical-scribing, 2024. Accessed 2025-05-21.

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[6] Jiwoong Sohn, Yein Park, Chanwoong Yoon, Sihyeon Park, Hyeon Hwang, Mujeen Sung, Hyunjae Kim, and Jaewoo Kang. Rationale-guided retrieval augmented generation for medical question answering. *arXiv preprint arXiv:2411.00300*, 2024.

[7] Ivan Lopez, Akshay Swaminathan, Karthik Vedula, Sanjana Narayanan, Fateme Nateghi Haredasht, Stephen P Ma, April S Liang, Steven Tate, Manoj Maddali, Robert Joseph Gallo, et al. Clinical entity augmented retrieval for clinical information extraction. *npj Digital Medicine*, 8(1):45, 2025.

[8] Mercy Ranjit, Gopinath Ganapathy, Ranjit Manuel, and Tanuja Ganu. Retrieval augmented chest x-ray report generation using openai gpt models. In *Machine Learning for Healthcare Conference*, pages 650–666. PMLR, 2023.

[9] Nikhil J Dhinagar, Sophia I Thomopoulos, Emily Laltoo, and Paul M Thompson. Efficiently training vision transformers on structural mri scans for alzheimer's disease detection. In *2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 1–6. IEEE, 2023.

[10] Bernardo Silva, Jefferson Fontinele, Carolina Letícia Zilli Vieira, João Manuel RS Tavares, Patricia Ramos Cury, and Luciano Oliveira. Semi-supervised classification of dental conditions in panoramic radiographs using large language model and instance segmentation: A real-world dataset evaluation. *arXiv preprint arXiv:2406.17915*, 2024.

[11] Peter F Brown, Vincent J Della Pietra, Peter V Desouza, Jennifer C Lai, and Robert L Mercer. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–480, 1992.

[12] Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. Lstm neural networks for language modeling. In *Interspeech*, volume 2012, pages 194–197, 2012.

[13] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[14] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[15] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[16] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training.(2018), 2018.

[17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[18] xAI. Grok (chatbot). https://en.wikipedia.org/wiki/Grok_(chatbot), 2023. Accessed May 6, 2025.

[19] OpenAI. Introducing chatgpt. https://openai.com/blog/chatgpt/, Nov 2022. Accessed May 6, 2025.

[20] Google. Introducing gemini: our largest and most capable ai model. https://blog.google/technology/ai/google-gemini-ai/, 2023. Accessed May 6, 2025.

[21] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[22] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[23] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR, 2020.

[24] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781, 2020.

[25] Yucheng Shi, Shaochen Xu, Tianze Yang, Zhengliang Liu, Tianming Liu, Xiang Li, and Ninghao Liu. Mkrag: Medical knowledge retrieval augmented generation for medical question answering. In *AMIA Annual Symposium Proceedings*, volume 2024, page 1011, 2025.

[26] Gongbo Zhang, Zihan Xu, Qiao Jin, Fangyi Chen, Yilu Fang, Yi Liu, Justin F Rousseau, Ziyang Xu, Zhiyong Lu, Chunhua Weng, et al. Leveraging long context in retrieval augmented language models for medical question answering. *npj Digital Medicine*, 8(1):239, 2025.

[27] Liwen Sun, James Zhao, Megan Han, and Chenyan Xiong. Fact-aware multimodal retrieval augmentation for accurate medical radiology report generation. *arXiv preprint arXiv:2407.15268*, 2024.

[28] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[30] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[31] Wenhu Chen, Hexiang Hu, Xi Chen, Pat Verga, and William W Cohen. Murag: Multimodal retrieval-augmented generator for open question answering over images and text. *arXiv preprint arXiv:2210.02928*, 2022.

[32] Junjie Zhou, Zheng Liu, Shitao Xiao, Bo Zhao, and Yongping Xiong. Vista: visualized text embedding for universal multi-modal retrieval. *arXiv preprint arXiv:2406.04292*, 2024.

[33] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. In *Machine learning for healthcare conference*, pages 2–25. PMLR, 2022.

[34] Haotian Liu, Kilho Son, Jianwei Yang, Ce Liu, Jianfeng Gao, Yong Jae Lee, and Chunyuan Li. Learning customized visual models with retrieval-augmented knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15148–15158, 2023.

[35] Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. Uniir: Training and benchmarking universal multimodal information retrievers. In *European Conference on Computer Vision*, pages 387–404. Springer, 2024.

[36] Kai Zhang, Yi Luan, Hexiang Hu, Kenton Lee, Siyuan Qiao, Wenhu Chen, Yu Su, and Ming-Wei Chang. Magiclens: Self-supervised image retrieval with open-ended instructions. *arXiv preprint arXiv:2403.19651*, 2024.

[37] Zhuolin Yang, Wei Ping, Zihan Liu, Vijay Korthikanti, Weili Nie, De-An Huang, Linxi Fan, Zhiding Yu, Shiyi Lan, Bo Li, et al. Re-vilm: Retrieval-augmented visual language model for zero and few-shot image captioning. *arXiv preprint arXiv:2302.04858*, 2023.

[38] Abhijit Anand, Vinay Setty, Avishek Anand, et al. Context aware query rewriting for text rankers using llm. *arXiv preprint arXiv:2308.16753*, 2023.

[39] Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*, 2019.

[40] Mitko Gospodinov, Sean MacAvaney, and Craig Macdonald. Doc2query–: when less is more. In *European Conference on Information Retrieval*, pages 414–422. Springer, 2023.

[41] Chi-Min Chan, Chunpu Xu, Ruibin Yuan, Hongyin Luo, Wei Xue, Yike Guo, and Jie Fu. Rq-rag: Learning to refine queries for retrieval augmented generation. *arXiv preprint arXiv:2404.00610*, 2024.

[42] Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. M3docrag: Multi-modal retrieval is what you need for multi-page multi-document understanding. *arXiv preprint arXiv:2411.04952*, 2024.

[43] Hamin Koo, Minseon Kim, and Sung Ju Hwang. Optimizing query generation for enhanced document retrieval in rag. *arXiv preprint arXiv:2407.12325*, 2024.

[44] Reem Agbareia, Mahmud Omar, Ofira Zloto, Nisha Chandala, Tania Tai, Benjamin S Glicksberg, Girish N Nadkarni, and Eyal Klang. The role of prompt engineering for multimodal llm glaucoma diagnosis. *medRxiv*, pages 2024–10, 2024.

[45] Zheng-Zhe Zhan, Yu-Tao Xiong, Chen-Yuan Wang, Bao-Tian Zhang, Wen-Jun Lian, Yu-Min Zeng, Wei Liu, Wei Tang, and Chang Liu. Utilizing gpt-4 to interpret oral mucosal disease photographs for structured report generation. *Scientific Reports*, 15(1):5187, 2025.

[46] Yutong Zhang, Yi Pan, Tianyang Zhong, Peixin Dong, Kangni Xie, Yuxiao Liu, Hanqi Jiang, Zihao Wu, Zhengliang Liu, Wei Zhao, et al. Potential of multimodal large language models for data mining of medical images and free-text reports. *Meta-Radiology*, 2(4):100103, 2024.

[47] Dana Brin, Vera Sorin, Yiftach Barash, Eli Konen, Benjamin S Glicksberg, Girish N Nadkarni, and Eyal Klang. Assessing gpt-4 multimodal performance in radiological image analysis. *European Radiology*, 35(4):1959–1965, 2025.

[48] OpenAI. Pricing - openai api. https://platform.openai.com/docs/pricing, 2025. Accessed: 2025-05-24.

[49] Anthropic. Pricing - claude. https://docs.anthropic.com/en/docs/about-claude/pricing, 2025. Accessed: 2025-05-24.

[50] xAI. Models & pricing. https://docs.x.ai/docs/models, 2025. Accessed: 2025-05-24.

[51] QwenLM. Qwen2.5-vl. https://github.com/QwenLM/Qwen2.5-VL, 2025. Accessed: 2025-05-24.

[52] DeepSeek. Pricing - deepseek api. https://api-docs.deepseek.com/quick_start/pricing, 2025. Accessed: 2025-05-24.

[53] Google AI. Gemini api pricing. https://ai.google.dev/gemini-api/docs/pricing, 2025. Accessed: 2025-05-24.

[54] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at trec-3. *Nist Special Publication Sp*, 109:109, 1995.

[55] Retrieval augmented generation techniques. https://gettectonic.com/retrieval-augmented-generation-techniques/. Accessed: 2025-05-21.

[56] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022.

[57] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[58] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[59] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[60] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. 2021.

[61] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

[62] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[63] Anthropic. Claude opus 4 vision. https://www.anthropic.com/claude/opus, 2024. Accessed: 2025-05-21.

[64] xAI. Grok 2 vision. https://x.com/xai/status/1868045132760842734, 2024. Accessed: 2025-05-21.

[65] OpenAI. Gpt-4o vision. https://platform.openai.com/docs/guides/images-vision?api-mode=responses, 2025. Accessed: 2025-05-21.

GenAI code of conduct for students (2024-2025)

Generative AI (GenAI) assistance tools can be used to generate text, image, code, video, music or combinations of these. It includes typical tools like (but this list is not limited to): ChatGPT, Google Gemini, MS Copilot, Midjourney, Claude.ai, Perplexity.ai, Dall-E, …

Student name: **Zhongbo Yao**

Student number: **r0864590**

Please indicate with "X" whether it relates to a course assignment or to the master thesis:

O This form is related to a **course assignment**.

Course name: .....

Course number: .....

X This form is related to **my Master thesis**.

Title Master thesis: Laying the Foundation for Multimodal RAG in Dental Imaging: Evaluating GPT-4o Vision on Oral Panoramic Radiographs

Promoter: Professor Peter Claes, Professor Reinhilde Jacobs

Daily supervisor: Soroush Baseri Saadi

Please indicate with "X":

O **I did not use** any GenAI assistance tool.

X **I did use** GenAI Assistance. In this case **specify which ones** (e.g. ChatGPT, ...): ChatGPT, Grok

	Name of the GenAI tool used *Please indicate with "X" (possibly multiple times)* *in which way you were using GenAI:*				
GenAI assistance used as/for:	ChatGPT & Grok				
Language assistance	X				
Search engine	X				
Literature search	X				
Short-form input assistance					
Generating programming code	X				
Generating new research ideas	X				
Generating blocks of text					
Other (specify):					

O **As a language assistant for reviewing or improving texts I wrote myself**

- *Code of conduct*: This use is similar to using spelling and grammar check tools, you do not have to refer to use of GenAI in the text.
  Be careful:
  - Using GenAI tools on texts you did not write yourself to cover up plagiarism (by paraphrasing original texts) is not allowed.

O **As a search engine to get information on a topic or to search for existing research on the topic**

- *Code of conduct*: This use is similar to e.g. a Google search or checking Wikipedia. If you then write your own text based on this information, you do not have to refer to use of GenAI in the text.
  Be careful:
  - Be aware that the output of the GenAI tool cannot be guaranteed as a 100% reliable source of information. The output may not be entirely correct and be limited due to the databases it uses. Knowledge evolves and may change over time, it may be that the database of the GenAI tool is not up to date.

O **For literature search**

- *Code of conduct*: This use is comparable to e.g. a Google Scholar search.
  Be careful:
  - Be aware that the output is restricted to the database it is built on. After this initial search, look for scientific sources and conduct your own analysis of the source documents. Interpret, analyse and process the information you obtained; verify it and don't just copy-paste it.
  - Be aware that some GenAI tools (like ChatGPT) may output no or wrong references. As a student you are responsible for further checking and verifying the absence or correctness of references; don't just copy-paste it.

  If you then write your own text based on this information, you do not have to refer to use of GenAI in the text.

O **For short-form input assistance**

- *Code of conduct*: This use is similar to e.g. Google docs powered by generative language models

O **To generate programming code**

- *Code of conduct*: Use of GenAI for coding should be explicitly allowed by the teacher. If used for coding, correctly mention the use of GenAI assistance and cite it.

O **To generate new research ideas**

- *Code of conduct*: Further verify in this case whether the idea is novel or not. It is likely that it is related to existing work, which should be referenced then.

O **To generate blocks of text**

- *Code of conduct*: Inserting blocks of text without quotes and a reference to GenAI assistance in your report or thesis is not allowed. According to Article 84 of the exam regulations in evaluating your work one should be able to correctly judge on your own knowledge, understanding and skills. In case it is really needed to insert a block of text from a GenAI tool, mention it as a citation by using quotes. But this should be kept to an absolute minimum. When you literally copy elements from a conversation with a GenAI tool: Quote between quotation marks and refer according to the specified reference style or as a personal communication within the text itself. Describe the use of the GenAI-tool (tool name, version, date, ...) in the method section (if there is one) and optionally add the (link to the) full conversation as an attachment.

O **Other**

- *Code of conduct*: Contact the professor of the course or the promoter of the thesis. Inform also the program director. Motivate how you comply with Article 84 of the exam regulations. Explain the use and the added value of ChatGPT or another AI tool: ….

Further important guidelines and remarks

- GenAI assistance cannot be used related **to data or subjects under Non-Disclosure Agreement.**
- GenAI assistance cannot be used related **to sensitive or personal data due to privacy issues**.
- **Take a scientific and critical attitude** when interacting with GenAI assistance and interpreting its output. Don't become emotionally connected to AI tools.
- As a student you are responsible for complying with Article 84 of the exam regulations: your report or thesis should reflect your own knowledge, understanding and skills. Be aware that plagiarism rules also apply to (work that is the result of) the use of GenAI assistance tools.
- **Exam regulations Article 84**: "Every conduct individual students display with which they (partially) inhibit or attempt to inhibit a correct judgement of their own knowledge, understanding and/or skills or those of other students, is considered an irregularity which may result in a suitable penalty. A special type of irregularity is plagiarism, i.e. copying the work (ideas, texts, structures, designs, images, plans, codes , ...) of others or prior personal work in an exact or slightly modified way without adequately acknowledging the sources. Every possession of prohibited resources during an examination (see article 65) is considered an irregularity."
- In order to maintain academic integrity and avoid plagiarism **more information about being transparent on the use of GenAI assistance and about citing and referencing GenAI** can be found on this website for students (Dutch/English).
- **Additional reading : KU Leuven guidelines on responsible use of Generative AI tools, and other information** (Dutch/English)

A few final words

If you are uncertain whether or not you should declare your use of AI tools, we suggest that you discuss the matter with your teacher or promoter. It is safer to declare AI use when it is not needed than to withhold that declaration when it is required.

Finally, remember that advanced AI tools are new and that they can do things they could not do, up until recently – so we do not have all the answers about how to use them responsibly yet. It is important to follow-up on most recent evolutions in AI technologies, to be a bit cautious, to communicate with teachers, teachings assistants, supervisors, promoters and peers with open minds, to be as transparent as we can, and to learn together as we move along.

This code of conduct can be used as a framework for academic year 2024-2025, changes will be made based on new evolutions.