TNM Cancer Staging with Large Language Models: Comparative Analysis of MedPrompt and Other Structured Prompting Techniques

Rodrigo Stall Sikora, Gabriel Lino Garcia, João Paulo Papa

Department of Computing, São Paulo State University, Brazil

Abstract

TNM cancer staging is essential for assessing cancer severity, guiding treatment decisions, and predicting clinical outcomes. This standardized system allows healthcare professionals to assess tumor progression and tailor treatment strategies accordingly. Although large language models (LLMs) show promise in automating TNM staging, their clinical reliability depends on strict adherence to oncological guidelines. This study evaluates the performance of three LLMs—GPT-4-o mini, LLaMA 3.3 70B Instruct, and DeepSeek-R1-Distill-LLaMA-70B—in classifying TNM stages across 1,000 TCGA pathology reports from 33 cancer types. We compare traditional prompting techniques, including Zero-Shot and Zero-Shot with Chain-of-Thought, with the specialized MedPrompt and two novel methodologies: Approach A, which mimics the step-by-step reasoning process of medical professionals using self-generated staging rules, and Approach B, which follows the same structured process while explicitly integrating AJCC guidelines. Our findings show that Approach B achieved state-of-the-art results in the N and M categories, with N macro average precision reaching 0.88 and M macro average precision reaching 0.91. Additionally, MedPrompt achieved the highest mean macro average precision across the three TNM categories, with a score of 0.867. These findings highlight the critical role of domain-specific structured prompting in improving LLM accuracy, minimizing hallucinations, and ensuring clinical reliability in automated TNM cancer staging.

Keywords: LLM, TNM Cancer Staging, GPT-4-o mini, LlaMa 3.3 70B Instruct, DeepSeek-R1, Medprompt, AJCC Guidelines, Self-Consistency, Prompting Techniques, Zero-Shot learning, Few-Shot Learning, Chain-of-Thought, Medprompt, Clinical Reasoning, Generative AI, Medical AI, Natural Language Processing, NLP, Healthcare.

1. Introduction

Cancer staging classification is a fundamental aspect of oncology, offering a systematic approach to assessing the severity and extent of cancer in a patient's body [6]. One of the most widely used systems is the TNM staging system, developed by the American Joint Committee on Cancer (AJCC). This system categorizes cancer based on three key parameters:

Tumor (T): Describes the size and extent of the primary tumor, ranging from T0 (no evidence of a tumor) to T4 (large or invasive tumors).

Nodes (N): Indicates the degree of regional lymph node involvement, from N0 (no lymph node involvement) to N3 (extensive involvement).

Metastasis (M): Assesses whether cancer has spread to distant sites, with M0 indicating no metastasis and M1 signifying the presence of distant metastases.

For instance, a breast cancer case classified as T2N1M0 corresponds to a tumor measuring 2 to 5 cm (T2), involvement of one axillary lymph node (N1), and no distant metastasis (M0). Importantly, different cancer types follow distinct staging rules, as TNM classifications are tailored to account for variations in tumor biology, anatomical sites, and prognostic factors. The TNM staging system serves as a standardized framework essential for treatment planning, prognosis assessment, and effective communication among healthcare providers. Given its direct impact on treatment decisions and patient outcomes, ensuring the accuracy of TNM staging is critical. However, staging is often complex, requiring the interpretation of diverse clinical and pathological information while adhering to strict classification guidelines.

To address these challenges, artificial intelligence (AI) and natural language processing (NLP) have emerged as promising tools for improving cancer staging classification. Large language models (LLMs), have demonstrated remarkable capabilities in understanding and analyzing medical data [9, 11]. By leveraging vast amounts of clinical knowledge, these models can assist in extracting relevant information and ensuring consistency in TNM staging assessments. However, applying LLMs to structured medical classification tasks requires careful guidance to ensure reliability and accuracy.

In recent years, large language models have revolutionized the field of natural language processing [11]. Advanced models like GPT-4-o, LLaMa 3 and DeepSeek-R1, are capable of understanding, generating, and analyzing complex medical data with remarkable accuracy [9].

A key methodology for enhancing LLM performance in medical applications is the use of prompting techniques. Prompting involves providing the model with specific instructions or

examples to guide its responses. Techniques such as Zero-shot, Few-shot, Chain-of-Thought reasoning, Self-Consistency, and MedPrompt have been shown to improve the precision and reliability of cancer staging classification [1, 3, 4].

This paper investigates how various prompting techniques can enhance the accuracy of TNM staging classification using LLMs across a wide range of cancer types. While most studies concentrate on one or a few specific cancer types, this research takes a broader approach, utilizing the TCGA dataset, covering 33 cancer types. We compare traditional prompting methods—such as Zero-shot, Chain-of-thought, and MedPrompt with two innovative approaches: Approach A and Approach B. These methods integrate structured reasoning and domain-specific staging rules to improve classification precision. Our objective is to identify the most reliable techniques for TNM staging predictions, with the ultimate goal of advancing clinical decision-making.

2. Related Works

The application of NLP to TNM cancer staging has evolved significantly over the years. Before the rise of OpenAI's ChatGPT in 2022, early NLP approaches primarily relied on rule-based systems and pattern matching, laying the groundwork for later advancements. However, these methods often struggled with generalizability and flexibility in processing unstructured clinical text.

In 2021, researchers from Weill Cornell Medicine introduced the Leo NLP System [7], an advanced model designed to analyze medical data and identify TNM classifications for prostate, breast, and colorectal cancers. The system demonstrated high accuracy for these cancer types, with its effectiveness potentially extending to other cancer types. However, a key limitation of Leo NLP System was its reliance on explicit mentions of TNM classifications in the text. For example, it could recognize a tumor as T2 if explicitly stated, but it struggled to infer classifications from contextual information, such as tumor size or lymph node involvement. This reliance on explicit mentions severely limited its ability to stage cancers when the TNM classifications were implied rather than directly mentioned.

In 2022, Park et al. [10] developed a deep-learning NLP model aimed at automating the extraction of lung cancer staging information from unstructured PET-CT radiology reports. The model, trained on over 20,000 reports, combined convolutional and recurrent neural networks with pseudo-labeling techniques to classify primary tumor sites, metastatic lymph nodes, and distant metastases. For nodal staging (N-category), the model achieved an overall accuracy of 0.7663, demonstrating strong performance. For metastatic staging (M-category), the model achieved an accuracy of 0.615. This study showcased the potential of deep-learning NLP approaches to automate cancer staging, reduce the need for manual annotations, and facilitate large-scale clinical research.

In contrast to traditional NLP systems that rely on predefined rules or pattern matching, large language models provide a more adaptable approach to TNM Cancer Staging. By leveraging deep contextual reasoning, LLMs can infer TNM classifications from unstructured clinical narratives, even when explicit labels are absent. These models process entire pathology reports holistically, extracting implicit relationships between tumor size, lymph node involvement, and metastasis status. Furthermore, LLMs generalize across different report formats, terminologies, and writing styles, making them well-suited for real-world clinical documentation.

LLMs have gained significant traction in medical applications, demonstrating advances in clinical decision support, medical image interpretation, drug discovery, and patient outcome prediction [11]. By analyzing vast amounts of medical data—such as patient records, clinical notes, and diagnostic images—LLMs enhance diagnostic accuracy, improve treatment predictions, and identify novel therapeutic pathways. Additionally, they assist healthcare professionals by providing evidence-based recommendations, facilitating personalized treatment plans, and automating routine tasks.

Despite the growing adoption of LLMs in medical fields, their application to TNM Cancer Staging remains relatively limited. Pioneering work by Chang et al. [1, 2] established a foundation for leveraging open-source LLMs in this domain, demonstrating their potential for accurate cancer stage classification. This study uses the TCGA dataset, comprising 33 cancer types, to evaluate LLM performance in TNM staging with a more diverse dataset. Table 1 compares the macro-average precision of the T, N, and M categories across various models, based on the results from Chang et al. [1, 2], illustrating the performance of different techniques.

Table 1: Macro-average precision for the T, N, and M categories on the TCGA dataset across various models and techniques, as reported by Chang et al. [1, 2].

Id	Model	Technique	T	N	M
1	Llama-2-70b-chat	Zero-Shot	0.84	0.63	0.54
2	Llama-2-70b-chat	Zero-Shot + Chain-of-Thought	0.82	0.69	0.55
3	Llama-2-70b-chat	Few-Shots	0.74	0.61	0.51
4	ClinicalCamel-70B	Zero-Shot	0.79	0.83	0.54
5	ClinicalCamel-70B	Zero-Shot + Chain-of-Thought	0.78	0.84	0.55
6	ClinicalCamel-70B	Few-Shots	0.66	0.78	0.52
7	Med42-70B	Zero-Shot	0.81	**0.88**	0.55
8	Med42-70B	Zero-Shot + Chain-of-Thought	0.80	0.84	0.55
9	Med42-70B	Few-Shots	0.74	0.82	**0.62**
10	Med42-70B	Zero-Shot + Chain-of-Thought + Self-Consistency	**0.865**	0.868	-
11	Med42-70B	Ensemble Reasoning (EnsReas)	0.86	0.875	-

In 2023, Hidetoshi Matsuo et al. [8] investigated the use of LLMs for TNM cancer staging in lung cancer radiology reports. They compared a Zero-Shot approach with an instruction-tuned approach, using a prompt specifically designed for TNM classification of lung cancer. Their study utilized a dataset of 162 chest CT reports, documented by board-certified radiologists, along with their corresponding ground-truth TNM classifications. Using GPT-3.5-turbo, the instruction-tuned approach showed an average accuracy improvement of 0.12 across all categories over the Zero-Shot approach. Table 2 presents a comparison of the accuracy results between the two approaches.

Table 2: Accuracy of TNM Staging using Zero-Shot and Instruction-Tuned prompts on a dataset of 162 chest CT reports by Hidetoshi Matsuo et al. [8].

Id	Model	Technique	T	N	M
1	GPT3.5-turbo	Zero-Shot	0.29	0.70	0.86
2	GPT3.5-turbo	Instruction-Tuned	0.47	0.80	0.94

Harsha Nori et al. [3] demonstrated that advanced prompting techniques can significantly enhance LLM accuracy in medical tasks. In 2023, they introduced MedPrompt, a structured and multi-faceted prompting strategy that improves LLM performance in medical reasoning tasks. MedPrompt combined several key techniques, including K-Nearest Neighbors (KNN) for retrieving relevant few-shot examples, Chain-of-Thought reasoning to improve step-by-step logical deductions, and ensemble methods with choice shuffling to reduce bias and enhance robustness. When applied to the MedQA dataset, GPT-4 achieved an accuracy of 81.7% in a zero-shot setting, which increased to 90.2% with MedPrompt. This 8.5% improvement underscores how carefully designed prompting strategies can enhance LLM reasoning in complex medical tasks. Given MedPrompt's success in medical reasoning, it shows promise for improving LLM-based TNM cancer staging, where the model must understand nuanced relationships between various staging parameters.

Karan Singhal et al. [4] emphasized the impact of individual prompting techniques—such as Few-Shot prompting, Chain-of-Thought, Self-Consistency, and Ensemble Refinement—in improving LLM performance. These techniques have been successfully applied in various medical domains, but their impact on cancer staging classification remains underexplored. This paper aims to fill this gap by systematically evaluating how these advanced prompting strategies can enhance the accuracy of LLM-based TNM staging classification.

3. Material and Methods

This section outlines the dataset used for evaluation, the Large Language Models selected for testing, and the prompting techniques applied in developing and assessing the cancer staging classification models.

3.1. Data

The dataset used in this study was The Cancer Genome Atlas (TCGA), which contains clinical data from over 11,000 cancer patients across 33 different cancer types. This is a public dataset which can be access thought github (https://github.com/tatonetti-lab/tcga-path-reports). From this dataset, 9,523 clinical cases descriptions were identified, of which 3,907 included complete TNM cancer classifications across all three categories (T, N, and M).

To address class imbalance, 1,000 clinical cases with complete TNM classifications were selected using a stratified sampling approach. This method oversampled underrepresented categories to ensure a more balanced distribution of T, N, and M classifications. The distributions of the T, N, and M categories in both the full dataset and the selected sample are provided in Tables 3, 4, and 5.

Table 3: Distribution of T Classifications in the Selected Sample from the TCGA Dataset

Classification	Total Count	Percentage	Selected Count	Selected Percentage
T2	1127	28.8%	193	19.3%
T3	972	24.9%	236	23.6%
T1	388	9.9%	56	5.6%
T1b	200	5.1%	41	4.1%
T4a	174	4.5%	81	8.1%
T4	154	3.9%	70	7.0%
T2a	143	3.7%	33	3.3%
T3a	140	3.6%	49	4.9%
T1c	125	3.2%	40	4.0%
T4b	120	3.1%	45	4.5%
T1a	105	2.7%	34	3.4&
T2b	98	2.5%	34	3.4%
T3b	81	2.1%	35	3.5%
T1b1	39	1%	23	2.3%
T1b2	15	0.4%	12	1.2%
TX	8	0.2%	6	0.6%
T2a2	5	0.1%	4	0.4%
T2a1	4	0.1%	3	0.3%
T3c	3	0.1%	3	0.3%
T4d	3	0.1%	1	0.1%
T0	1	<0.1%	0	0%
T4c	1	<0.1%	0	0%
T4e	1	<0.1%	1	0.1%
Total	3907	100%	1000	100%

Table 4: Distribution of N Classifications in the Selected Sample from the TCGA Dataset

Classification	Total Count	Percentage	Selected Count	Selected Percentage
N0	2263	57.9%	388	33.8%
N1	727	18.6%	182	18.2%
N2	340	8.7%	95	9.5%
N1a	198	5.1%	69	6.9%
N1b	82	2.1%	55	5.5%
N3	78	2%	54	5.4%
N3a	75	1.9%	56	5.6%
N2a	63	1.6%	43	4.3%
N2b	50	1.3%	38	3.8%
N2c	20	0.5%	13	1.3%
N1c	5	0.1%	3	0.3&
N3b	5	0.1%	3	0.3%
N3c	1	<0.1%	1	0.1%
Total	3907	100%	1000	100%

Table 5: Distribution of M Classifications in the Selected Sample from the TCGA Dataset

Classification	Total Count	Percentage	Selected Count	Selected Percentage
M0	3657	93.6%	848	84.8%
M1	222	5.7%	134	13.4%
M1a	19	0.5%	13	1.3%
M1b	8	0.2%	5	0.5%
M1c	1	<0.1%	0	0%
Total	3907	100%	1000	100%

3.2. Large Language Model

For this experiment, three large language models (LLMs) were selected based on their performance, accessibility via APIs, and suitability for integration into clinical applications: GPT-4-o mini, LLaMA 3.3 70B Instruct, and DeepSeek-R1-Distill-LLaMA-70B.

GPT-4-o mini strikes a balance between cost-efficiency and strong performance across various tasks. Introduced in July 2024, it is OpenAI's most affordable small-scale model, yet it still demonstrates high accuracy on diverse datasets [5]. Despite its lower computational cost, GPT-4-o mini has shown notable capabilities, making it a suitable choice for this study. Notably, like GPT-4, GPT-4-o mini was not fine-tuned on medical data, which underscores its generalization ability for tasks like cancer staging without the need for specialized training.

LLaMA 3.3 70B Instruct is part of the LLaMA family of models, released in 2024, and known for their strong performance and open-source availability. These models are designed for a wide range of natural language processing tasks, making them particularly versatile for both research and commercial environments. The 70B variant, with its large number of parameters, is capable of handling complex tasks, such as medical text classification, and has shown competitive results on various benchmarks [13]. Although not specifically fine-tuned for medical applications, its versatility and open-source nature make it an attractive candidate for clinical use.

DeepSeek-R1-Distill-LLaMA-70B is a distilled version of DeepSeek-R1, built on the LLaMA 3.3 70B base. It undergoes reinforcement learning (RL) to enhance reasoning capabilities, followed by supervised fine-tuning (SFT) for further refinement. This model outperforms similar architectures on multiple benchmarks, particularly excelling in chain-of-thought reasoning and self-verification. Designed to be a computationally efficient alternative to larger models, it maintains strong performance in complex problem-solving, mathematical reasoning, and code generation. Additionally, its open-source availability makes it a valuable asset for the research community, pushing the boundaries of distilled, high-performance LLMs [12].

In summary, each of these models offers unique strengths: GPT-4-o mini provides cost-effective generalization, LLaMA 3.3 70B Instruct offers versatility across tasks, and DeepSeek-R1-Distill-LLaMA-70B combines advanced reasoning capabilities with efficiency. These strengths make them well-suited for clinical applications, and their selection for this study aims to leverage these features to improve clinical reasoning and decision-making.

3.3. Prompting Techniques

In this experiment, several well-known prompting techniques were utilized. Additionally, I introduced Approach A and Approach B, designed to emulate the steps a oncologist follows to perform a TNM cancer classification. These techniques are described below:

Zero-Shot: This technique involves providing the model with a prompt that contains no examples or prior context related to the task. The model relies solely on its general knowledge to generate a response based on the prompt.

Few-Shot: Few-Shot prompting provides the model with a small number of examples within the prompt to illustrate the task at hand. This approach helps the model understand the format and expectations of the desired output.

Random Few-Shot: A variation of Few-Shot prompting, where examples are selected randomly from a set of available examples.

KNN Few-Shot: KNN Few-Shot integrates Few-Shot learning with the k-nearest neighbors (KNN) algorithm to refine the selection of examples used for inference. This method identifies and selects the most similar examples from a training set based on their proximity to the current input. By focusing on closely related examples, KNN Few-Shot enhances the model's ability to make relevant, contextually accurate predictions, improving performance on tasks with limited labeled data.

Self-Generated Chain-of-Thought: This method involves the model generating its own intermediate reasoning steps or chain of thought based on the prompt, rather than following a predefined structure. It allows the model to create a logical progression of ideas tailored to the specific query, potentially leading to more accurate and contextually relevant responses.

Self-Consistency: Self-Consistency is an ensemble approach where multiple models generate responses to the same prompt, and the most frequent answer is selected as the final output. This method improves the reliability of the model by leveraging the diversity of responses. The model is typically run with varying temperature settings, which leads to different potential responses. By identifying the most consistent answer from these multiple outputs, Self-Consistency helps reduce the influence of outliers and ensures a more accurate and dependable result.

Ensemble Refinement: Introduced by Harsha Nori et al. [4], Ensemble Refinement (ER) builds on chain-of-thought and self-consistency. ER involves a two-stage process: first, the model generates multiple possible outputs through temperature sampling, which includes both an explanation and an answer for a given task. Then, the model is conditioned on these previous generations, as well as the original prompt, and is prompted to refine its explanation and answer. This approach aggregates over the generated answers rather than simply selecting the most frequent answer, which allows the model to evaluate the strengths and weaknesses of its own explanations. ER is particularly useful in multiple-choice settings, where it improves the overall performance by refining responses based on various generated answers. The model performs multiple rounds of this refinement process, ultimately determining the final output through a plurality vote. This technique enhances accuracy by allow-

ing the model to condition on its own prior outputs, providing a more reliable and coherent answer.

MedPrompt: MedPrompt is an advanced prompting framework specifically designed to enhance the performance of large language models in medical tasks. It integrates a combination of state-of-the-art techniques, including K-Nearest Neighbors (KNN) Few-Shot learning, Self-Generated Chain-of-Thought reasoning, and Self-Consistency [3].

At its core, MedPrompt adapts the model's input to leverage both contextual and example-driven learning. It begins by embedding relevant examples within the prompt, utilizing the KNN approach to select instances most similar to the current task, ensuring the model is informed by contextually relevant data. The integration of Chain-of-Thought allows the model to articulate reasoning step-by-step, improving interpretability and the generation of logically coherent outputs. The method further refines predictions by employing Self-Consistency, which aggregates multiple outputs to identify the most consistent and reliable response. Figure 1 displays the Medprompt components and additive contributions to performance on the MedQA benchmark.

A notable aspect of MedPrompt is its use of options shuffling, which dynamically adjusts the prompt by altering the structure of the provided options to maximize the model's ability to select the correct prediction. This enhances accuracy, particularly in tasks with complex or nuanced medical reasoning, by encouraging the model to process and evaluate information from different angles.

The combined effect of these strategies results in a significant improvement in the accuracy and reliability of medical predictions, where precision is crucial. MedPrompt's holistic approach, which integrates both data-driven and reasoning-based techniques, enhances the LLM's efficiency in generating more contextually appropriate, accurate, and consistent responses in medical settings.

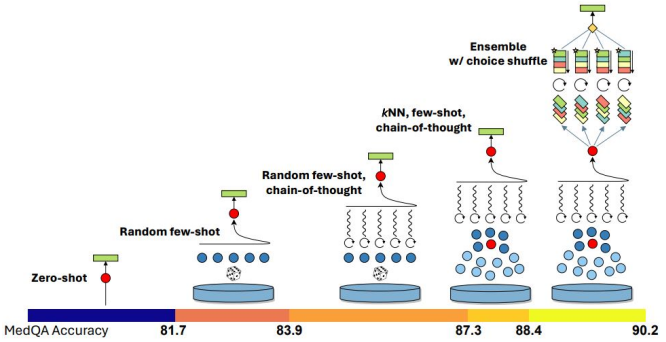

Figure 1: Visual illustration of Medprompt components and additive contributions to performance on the MedQA benchmark. The prompting strategy combines kNN-based few-shot example selection, LLM generated chain-of-thought prompting, and answer-choice shuffled ensembling. Relative contributions of each component are shown at the bottom. Image taken from [3].

3.4. Innovative Prompting Techniques

This paper introduces two prompting techniques for TNM Cancer Staging, Approach A and Approach B.

Approach A: This approach aims to replicate the steps commonly followed by an oncologist in classifying a TNM cancer stage. The method proceeds sequentially, starting with identifying the cancer location, followed by applying the AJCC Staging Rules specific to that location. The process consists of the following steps:

Prompt Initialization: The LLM is provided with a general prompt containing information about TNM cancer staging to establish context.

Cancer Location Identification: The model is asked to identify the cancer location in the patient on the clinical report.

Staging Rule Application (Self-Generated): The LLM then self-generates the AJCC Staging Rules relevant to the identified cancer location. The model relies on its internal knowledge and reasoning abilities to derive the appropriate staging rules, without external references.

Chain-of-Thought Generation: The model generates a Chain-of-Thought to reason through the staging process, ensuring each classification step is logically justified.

Classification Generation: Finally, the model produces the T (Tumor), N (Nodes), and M (Metastasis) classifications based on the earlier steps, providing a comprehensive TNM staging result.

This approach is designed to simulate the cognitive process of an oncologist, following a structured methodology to derive accurate and contextually appropriate TNM classifications. Figure 2 illustrate the Approach A.

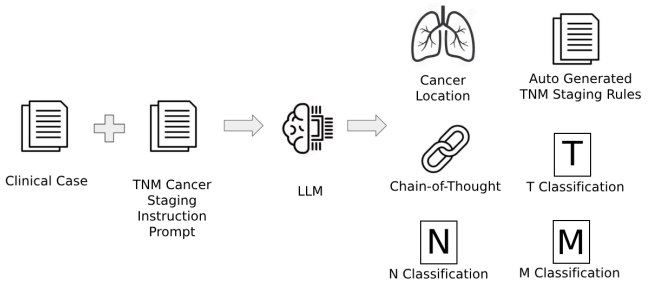

Figure 2: Visual illustration of Approach A to Cancer Staging with LLMs. This approach replicates the step-by-step reasoning process a pathologist follows when determining TNM cancer staging. It begins by identifying the tumor's location, as different cancer types follow distinct staging criteria. Next, it applies auto-generated TNM staging rules specific to that cancer type, ensuring that the classification aligns with established guidelines while leveraging automation for consistency and efficiency. Finally, it systematically evaluates tumor size, lymph node involvement, and metastasis status, reasoning through each factor before assigning the T, N, and M classifications. This structured approach enhances accuracy by mirroring expert decision-making.

Approach B: In this approach, we replicate the steps typically used by oncologist for TNM cancer staging, similar to Approach A, but with a key distinction: instead of generating the AJCC Classification Staging Rules, we provide them directly to the model. This ensures greater control over the staging criteria. Additionally, the LLM is tasked with summarizing the clinical case, identifying tumor size, reasoning through each TNM stage, generating a chain of thought, and ultimately assigning the TNM classifications. Figure 3 illustrates this approach. The

approach follows these structured steps:

Cancer Category Classification: A LLM classifier first identifies the cancer location based on the categories of the AJCC Cancer Staging Manual [6].

Staging Rules Input: Once the cancer location is determined, the LLM receives the corresponding AJCC staging rules. This ensures precise control over the criteria used for staging and mitigates the risk of hallucinations from self-generated rules, present in Approach A or any other technique discussed in this paper.

Few-Shot Examples Input: In addition to the TNM staging rules, the model is provided with few-shot examples specific to the identified cancer type. These examples help guide the LLM's reasoning by illustrating correct staging decisions in similar cases.

Clinical Case Analysis: The LLM then performs the following tasks:

1) Generate a summary of the clinical case.
2) Identify tumor size.
3) Provide reasoning for the T stage classification.
4) Provide reasoning for the N stage classification.
5) Provide reasoning for the M stage classification.
6) Generate a self-guided Chain-of-Thought.
7) Assign the final T classification.
8) Assign the final N classification.
9) Assign the final M classification.

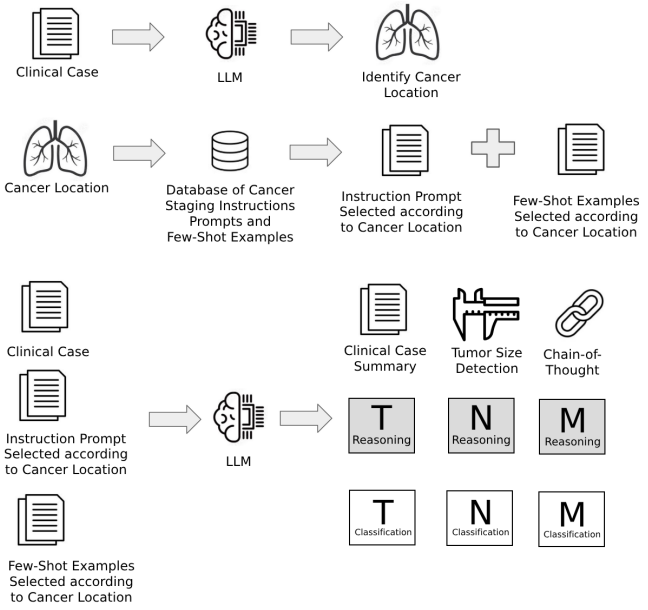

Figure 3: Visual illustration of Approach B to Cancer Staging with LLMs. This 3 step approach replicates the structured reasoning process pathologists use for TNM cancer staging while ensuring greater control over the classification criteria. It begins by identifying the cancer type using an LLM classifier, as different cancers follow distinct staging guidelines. Next, instead of generating TNM staging rules, the model is provided with predefined AJCC staging rules specific to that cancer type, reducing the risk of inaccuracies. Additionally, few-shot examples tailored to the cancer type are incorporated to guide the model's reasoning. Finally, the LLM systematically analyzes the clinical case—summarizing key findings, identifying tumor size, and providing detailed reasoning for the T, N, and M classifications. A self-guided Chain-of-Thought is used to enhance interpretability before assigning the final TNM classifications. This structured approach mirrors expert decision-making while leveraging controlled inputs for improved reliability.

4. Results and Discussion

4.1. Comparative Evaluation of Classification Techniques

To effectively compare the different techniques used in the TNM Cancer Classification process, the 1,000 samples from the dataset were processed using each technique, and their macro-average precision was inferred for each category, among other metrics.

Figure 4 compares the T macro average precision of three LLMs across six prompting techniques. For this category, LLaMa 3.3 70B Instruct achieved the highest performance (0.84) with MedPrompt, though its gain over Zero-Shot is marginal (+0.02), suggesting a strong baseline. GPT-4o mini benefits the most from prompting, improving from 0.73 (Zero-Shot) to 0.83 (+0.10) with MedPrompt. In contrast, DeepSeek-R1-Distill-LLaMa-70B shows minimal improvement (+0.01). One of the reasons for this small improvement maybe be because its Zero-Shot approach already incorporates Chain-of-Thought reasoning, leaving little room for external prompting gains.

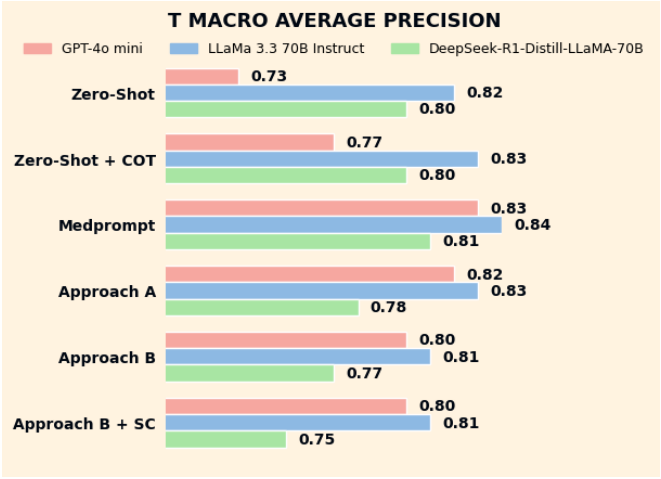

Figure 4: T Macro Average Precision over different techniques and models used for TNM Staging the TCGA Dataset.

Figure 5 presents the N macro average precision. LLaMa 3.3 70B Instruct achieved the highest score (0.88) using Approach B, though gains over Zero-Shot remain small (+0.02). GPT-4o mini sees a slight improvement (+0.02) with Approach B + Self-Consistency, indicating limited returns from advanced prompting. DeepSeek-R1-Distill-LLaMa-70B underperforms relative to the others but still benefits marginally from Approach B (+0.02).

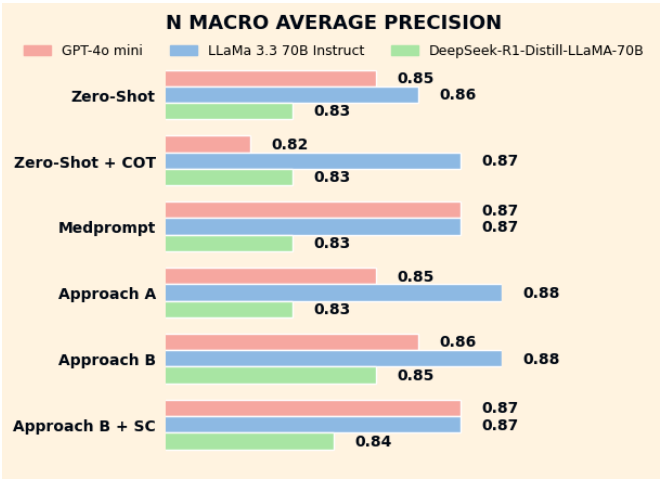

Figure 5: N Macro Average Precision over different techniques and models used for TNM Staging the TCGA Dataset.

Figure 6 compares the M macro average precision. GPT-4o mini achieved the highest score (0.91) with Approach B + Self-Consistency, improving by +0.06 over Zero-Shot. LLaMa 3.3 70B Instruct sees a substantial gain (+0.10) with MedPrompt. DeepSeek-R1-Distill-LLaMa-70B also performs well, reaching 0.90 with MedPrompt (+0.04 over Zero-Shot).

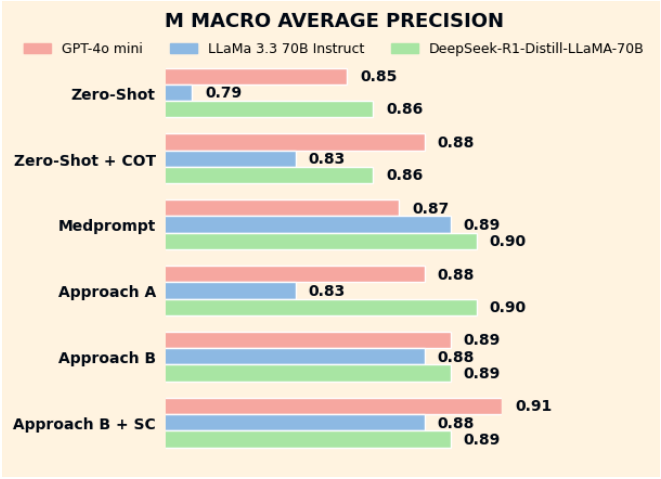

Figure 6: M Macro Average Precision over different techniques and models used for TNM Staging the TCGA Dataset.

Overall, these findings highlight the effectiveness of the prompting strategies in enhancing the model's macro-average precision. This is clearly demonstrated in Tables 6 and 7, which show the average improvement of each technique over the Zero-Shot baseline for GPT-4o mini and LLaMa 3.3 70B Instruct. DeepSeek-R1-Distill-LLaMa-70B was excluded from this comparison because its Zero-Shot baseline already incorporates Chain-of-Thought and reasoning approaches, making a direct comparison inappropriate.

Table 6: Average improvement in Macro Average Precision for each technique compared to the baseline of Zero-Shot for TNM Cancer Staging using GPT-4o mini

Technique	GPT-4o mini Improvement
Zero-Shot + CoT	+ 1.3%
Approach A	+ 3.7%
Approach B	+ 4.0%
Medprompt	+ 4.7%
Approach B + SC	+ 5.0%

Table 7: Average improvement in Macro Average Precision for each technique compared to the baseline of Zero-Shot for TNM Cancer Staging using LLaMa 3.3 70B Instruct

Technique	LLaMA 3.3 70B Instruct Improvement
Zero-Shot + CoT	+ 2.0%
Approach A	+ 2.3%
Approach B + SC	+ 3.0%
Approach B	+ 3.3%
Medprompt	+ 4.3%

4.2. Approach B + Self-Consistency Evaluation

Among the models tested and techniques employed, GPT-4o mini using Approach B with Self-Consistency demonstrated

the highest improvement in macro-average precision over the Zero-Shot baseline (+5%), achieving macro-average precisions of T: 0.80, N: 0.87, and M: 0.91. These results highlight the effectiveness of combining structured staging methodologies with consistency-enhancing techniques.

One of the key strengths of this approach is its structured methodology, which closely mirrors the process used by oncologists for TNM cancer staging. By explicitly providing the AJCC Classification Staging Rules instead of requiring the LLM to infer them, Approach B ensures greater control over the staging criteria and reduces the risk of hallucinations. Additionally, the inclusion of a clinical case summary, tumor size identification, reasoning for each TNM component, and a self-guided Chain-of-Thought (CoT) fosters a more rigorous and transparent decision-making process.

The integration of Self-Consistency further enhances the reliability of this approach by mitigating variability in LLM-generated responses. Instead of relying on a single inference, Self-Consistency enables multiple independent reasoning pathways, refining the final decision by selecting the most consistent outputs. This technique helps counteract occasional misclassifications and improves overall staging accuracy.

By structuring the staging process through defined rules, examples, and systematic reasoning, this approach minimizes the model's dependence on implicit knowledge and instead reinforces decision-making grounded in established medical guidelines. These findings emphasize the potential of structured LLM prompting techniques in clinical applications, supporting the broader adoption of AI-assisted cancer staging models.

The next section presents a detailed analysis of the evaluation results, further exploring the impact of this approach on classification accuracy and clinical reliability.

4.2.1. T Category

Figures 7 and 8 show the confusion matrices for the T category. In Figure 8, subcategories such as T1a, T1b, T1b1, T1b2, and T1c are clustered under T1. T2a, T2a1, T2a2, and T2b are clustered under T2, etc. Additionally, the T0 and TX categories were excluded from the clustered results for two reasons: (1) to evaluate the overall T category, and (2) to align with the benchmark [1, 2], which uses only the clustered categories and not the TNM Staging subcategories. To ensure comparability with the benchmark, Precision, Recall, F1-score, and Accuracy were computed using the clustered categories.

The confusion matrices illustrate the classification model's effectiveness, with most data concentrated along the diagonal. This indicates that the model correctly identifies instances in each category, leading to high classification accuracy. In an ideal confusion matrix, the predicted labels closely match the actual labels, with diagonal elements representing correct classifications. The strong presence of values along the diagonal suggests that the model performs well across different categories.

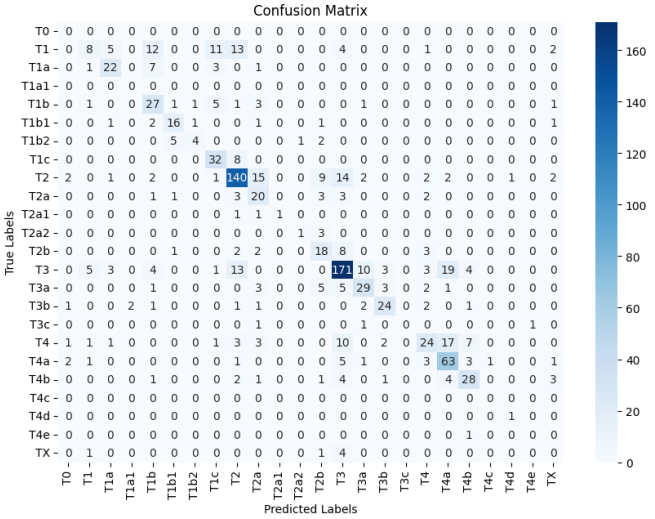

Figure 7: Confusion Matrix of the T category using GPT-4o mini with Approach B and Self Consistency.

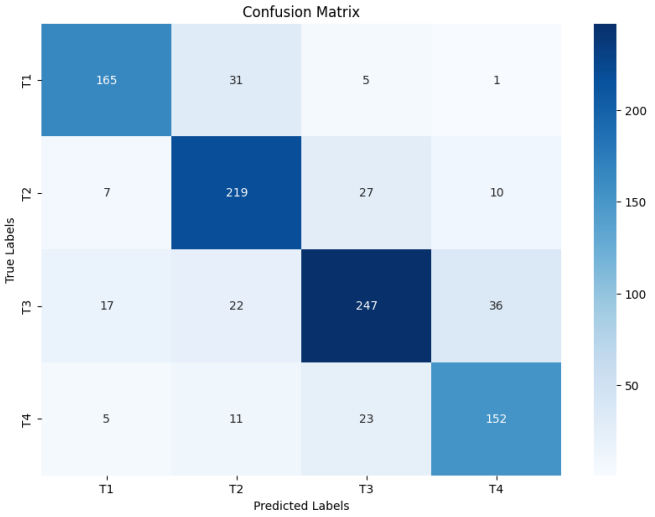

Figure 8: Confusion Matrix of the clustered T category using GPT-4o mini with Approach B and Self Consistency. TX and T0 were removed to facilitate the comparison with the benchmark.

Furthermore, the clustering of subcategories like T1a, T1b, T1b1, T1b2, and T1c under T1 strengthens the evaluation by grouping similar categories together, which allows for a more comprehensive analysis of the T category. This organization highlights the model's ability to generalize across the subcategories and makes it easier to assess its overall performance.

Table 8 presents the Precision, Recall, F1-score, and Accuracy for the T category over the selected data using this approach.

Table 8: Precision, Recall, F1-score, and Accuracy for the category T over the selected dataset using Approach B with Self-Consistency.

Class	Precision	Recall	F1-score	Support
T1	0.851	0.817	0.833	202
T2	0.774	0.833	0.802	263
T3	0.818	0.767	0.792	322
T4	0.764	0.796	0.779	191
T Macro Average	0.802	0.803	0.802	978
Accuracy	0.800			978

4.2.2. N Category

Figure 9 shows the confusion matrix for the N category.

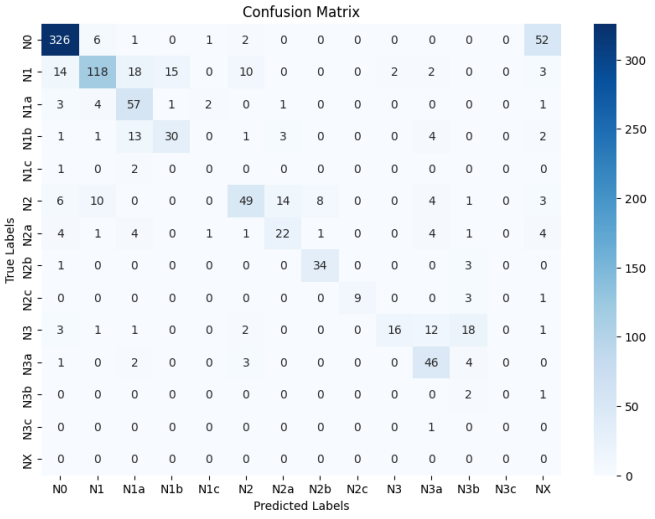

Figure 9: Confusion Matrix of the N category using GPT-4o mini with Approach B and Self Consistency.

Unlike the T category, the N category had many cases classified under the X subcategory (NX), even though their true label was actually N0. This may occur because in the TNM Cancer Staging system, the NX category indicates that the provided information was insufficient for classification, whereas the N0 category signifies the absence of lymph node metastasis. This misclassification could be due to incomplete or ambiguous pathology reports, where essential details confirming the absence of nodal metastasis were not explicitly stated, leading the model to default to NX rather than confidently assigning N0. In contrast, this issue may not have affected the T category as much because, given the nature of a cancer dataset, it is expected to contain an identifiable primary tumor, reducing uncertainty in classification between T0 and TX.

Figure 10 presents the confusion matrix for the N category with subcategories such as N1a, N1b, and N1c clustered under N1, etc.

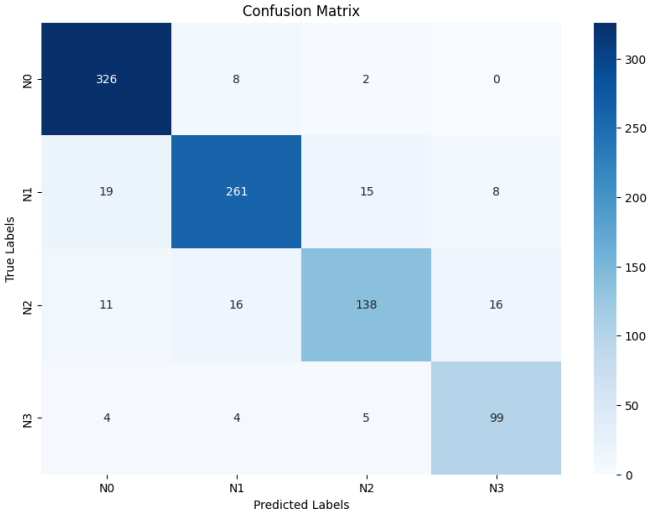

Figure 10: Confusion Matrix of the clustered N category using GPT-4o mini with Approach B and Self Consistency.

Table 9 presents the Precision, Recall, F1-score, and Accuracy for the N category over the selected data using Approach B with Self-Consistency.

Table 9: Precision, Recall, F1-score, and Accuracy for the category N over the selected dataset using GPT-4o mini and Approach B with Self-Consistency.

Class	Precision	Recall	F1-score	Support
N0	0.906	0.970	0.937	336
N1	0.903	0.861	0.882	303
N2	0.863	0.762	0.809	181
N3	0.805	0.884	0.843	112
N Macro Average	0.869	0.869	0.868	932
Accuracy	0.884			932

As seen in Table 9, the recall for categories N1, N2, and N3 is lower than that of category N0. This is expected, as classifying lymph node metastasis into N1, N2, or N3 is usually more challenging than simply categorizing it as N0, which implies the absence of lymph node metastasis.

4.2.3. M Category

Figure 11 shows the confusion matrix for the M category. Similar to the N classification, the model frequently misclassified M0 cases as MX. This misclassification may have the same underlying cause as the N0-to-NX errors, where MX, like NX, indicates that the provided information was insufficient to determine a definitive classification. In contrast, M0 signifies the absence of distant metastasis. This suggests that the pathology reports may have lacked explicit details confirming the absence of metastasis, leading the model to default to MX rather than confidently assigning M0.

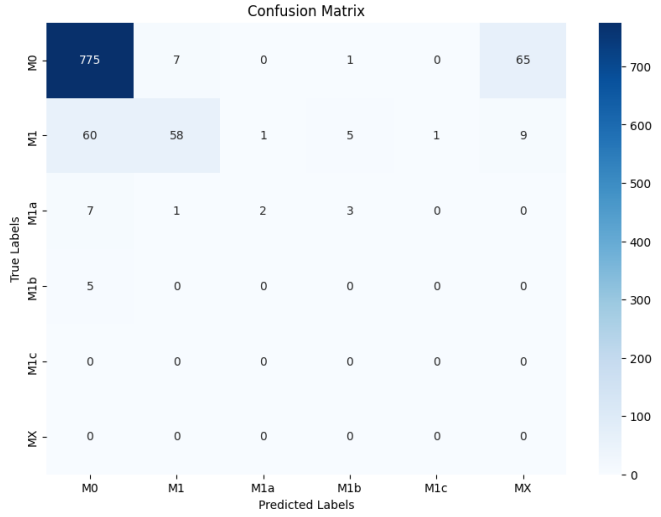

Figure 11: Confusion Matrix of the M category using GPT-4o mini with Approach B and Self Consistency.

Figure 12 presents the confusion matrix for the M category with subcategories such as M1a, M1b, and M1c clustered under M1.

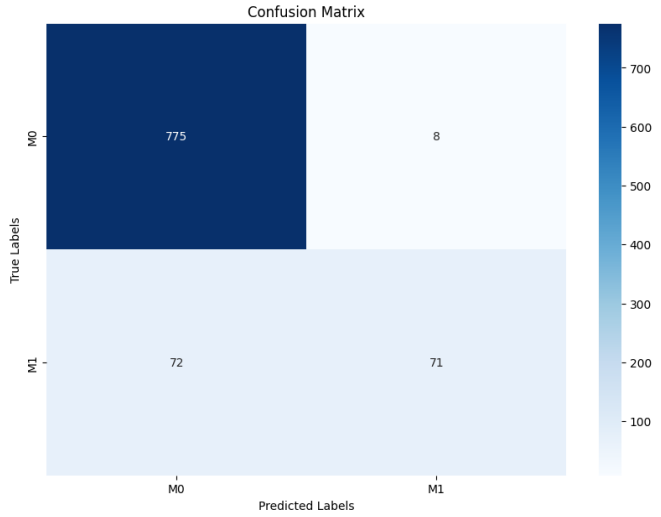

Figure 12: Confusion Matrix of the clustered M category using GPT-4o mini with Approach B and Self Consistency.

As seen in Table 10, the recall for the M1 category is significantly lower than for M0. This lower recall can be attributed to data imbalance, where the M0 category is more prevalent. In cases likes this with imbalanced data, precision can often be a better metric to evaluate the model.

Table 10: Precision, Recall, F1-score, and Accuracy for the category M over the selected dataset using GPT4-o mini with Approach B with Self-Consistency.

Class	Precision	Recall	F1-score	Support
M0	0.915	0.990	0.951	783
M1	0.899	0.497	0.640	143
M Macro Average	0.907	0.743	0.795	926
Accuracy	0.914			926

4.3. Benchmark Comparison

The body of research on the use of LLMs in Cancer Staging Classification is quite limited. This section aims to compare the findings from this experiment with the benchmark established by the studies by Chang et al., referenced in [1] and [2], which also used the TCGA dataset. Table 11 presents the Macro Average Precision for the T, N, and M classes, using various LLM models and techniques. The results from experiments with IDs 1 through 17 are discussed in this article, while those with IDs 18 through 28 are drawn from the studies conducted by Chang et al., as cited in references [1] and [2].

Table 11: Performance comparison of models using different prompting strategies. The values in the T, N, and M columns represent the macro-average precision for each one of this categories. For the models 24, 27 and 28, the M macro-average precision was not reported, and those values were left blank. The Average column represents the mean precision across the available categories.

Id	Model	Technique	T	N	M	Average
1	GPT-4o mini	Zero-Shot	0.73	0.85	0.85	0.810
2	GPT-4o mini	Zero-Shot + Chain-of-Thought	0.77	0.82	0.88	0.823
3	GPT-4o mini	Medprompt	0.83	0.87	0.87	0.857
4	GPT-4o mini	Approach A	0.82	0.85	0.88	0.850
5	GPT-4o mini	Approach B	0.80	0.86	0.89	0.850
6	GPT-4o mini	Approach B + Self-Consistency	0.80	0.87	**0.91**	0.860
7	Llama 3.3 70B Instruct	Zero-Shot	0.82	0.86	0.79	0.823
8	Llama 3.3 70B Instruct	Zero-Shot + Chain-of-Thought	0.83	0.87	0.83	0.843
9	Llama 3.3 70B Instruct	Medprompt	0.84	0.87	0.89	**0.867**
10	Llama 3.3 70B Instruct	Approach A	0.83	0.88	0.83	0.847
11	Llama 3.3 70B Instruct	Approach B	0.81	**0.88**	0.88	0.857
12	Llama 3.3 70B Instruct	Approach B + Self-Consistency	0.81	0.87	0.88	0.853
13	DeepSeek-R1-Distill-Llama-70B	Zero-Shot + Chain-of-Thought*	0.80	0.83	0.86	0.830
14	DeepSeek-R1-Distill-Llama-70B	Medprompt	0.81	0.83	0.90	0.847
15	DeepSeek-R1-Distill-Llama-70B	Approach A	0.78	0.83	0.90	0.837
16	DeepSeek-R1-Distill-Llama-70B	Approach B	0.77	0.85	0.89	0.837
17	DeepSeek-R1-Distill-Llama-70B	Approach B + Self-Consistency	0.75	0.84	0.89	0.827
18	Llama-2-70b-chat	Zero-Shot	0.84	0.63	0.54	0.670
19	Llama-2-70b-chat	Zero-Shot + Chain-of-Thought	0.82	0.69	0.55	0.686
20	Llama-2-70b-chat	Few-Shots	0.74	0.61	0.51	0.620
21	ClinicalCamel-70B	Zero-Shot	0.79	0.83	0.54	0.720
22	ClinicalCamel-70B	Zero-Shot + Chain-of-Thought	0.78	0.84	0.55	0.723
23	ClinicalCamel-70B	Few-Shots	0.66	0.78	0.52	0.653
24	Med42-70B	Zero-Shot	0.855	0.873	-	-
25	Med42-70B	Zero-Shot + Chain-of-Thought	0.80	0.84	0.55	0.730
26	Med42-70B	Few-Shots	0.74	0.82	0.62	0.726
27	Med42-70B	Zero-Shot + Chain-of-Thought + Self-Consistency	**0.865**	0.868	-	-
28	Med42-70B	Ensemble Reasoning (EnsReas)	0.86	0.875	-	-

DeepSeek-R1-Distill-Llama-70B inherently incorporates chain-of-thought reasoning in its architecture, meaning no explicit chain-of-thought prompting was necessary for the Zero-Shot experiment. However, since the model inherently applies this technique, labeling Experiment 13 as 'Zero-Shot' would be technically incorrect. Instead, the most accurate designation was 'Zero-Shot + Chain-of-Thought,' even though the prompt did not include explicit chain-of-thought instructions.

Figure 13 shows a Radar Chart for the Best Techniques by Model. The top techniques were selected based on the highest mean macro-average precision across the three categories (T, N, and M). For the Med42-70B model, where most experiments did not report the M category, the M category was excluded from the mean calculation.

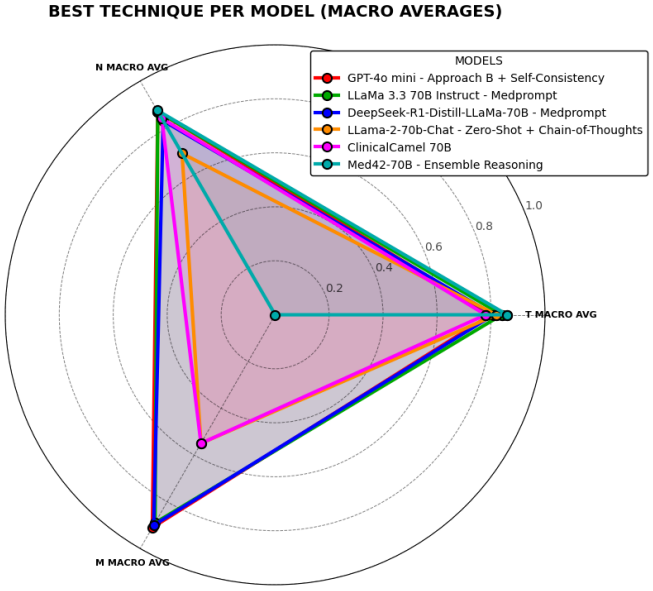

Figure 13: Radar Chart for Best Technique by Model. The radar chart illustrates the performance of four models—GPT-4o mini, LLaMA 3.3 70B Instruct, DeepSeek-R1-Distill-LLaMa-70B, Llama-2-70b-chat, ClinicalCamel-70B, and Med42-70B—across three key metrics: T Macro Average Precision, N Macro Average Precision, and M Macro Average Precision. Each model's optimal technique (e.g., Approach B with Self-Consistency for GPT-4o mini) is plotted to compare their precision in these dimensions.

Experiment 6, which used GPT-4o mini with Approach B and Self-Consistency, achieved state-of-the-art results with an M Macro Average Precision of 0.91. Experiment 9, using LLaMA 3.3 70B Instruct with MedPrompt, recorded the highest overall macro-average precision at 0.867, also setting a state-of-the-art result. Experiment 11, which used LLaMA 3.3 70B Instruct, achieved an N Macro Average Precision of 0.88, further advancing the state-of-the-art. The results in Table 11 and the radar chart highlight the effectiveness of Approach B in enhancing model precision, setting a new benchmark for TNM classification within the TCGA dataset.

5. Conclusion

This study highlights the potential of large language models (LLMs) in cancer staging classification. By comparing traditional prompting techniques such as Few-Shot, Chain-of-Thought, and MedPrompt with innovative approaches like Approach A and Approach B, we demonstrated that LLMs can significantly improve precision and consistency when classifying cancer stages from clinical reports. These structured techniques, which emulate medical decision-making processes, have proven particularly effective. Notably, Approach B, which incorporates explicit AJCC Staging Rules, enhanced reliability by reducing model hallucinations and increasing precision across various TNM categories.

Our findings suggest that with the right combination of prompting techniques, LLMs can handle specialized tasks like cancer staging with remarkable precision, opening new avenues for AI integration in clinical workflows. Future research should focus on refining these techniques, exploring specialized medical models, developing detailed instruction prompts, and expanding LLM applications to other areas of medical classification and diagnosis.

References

[1] Chang, C.H., Lucas, M.M., Lu-Yao, G., Yang, C.C.: Beyond Self-Consistency: Ensemble Reasoning Boosts Consistency and Accuracy of LLMs in Cancer Staging. arXiv (2024). https://doi.org/10.48550/arXiv.2404.13149

[2] Chang, C.H., Lucas, M.M., Lu-Yao, G., Yang, C.C.: Classifying Cancer Stage with Open-Source Clinical Large Language Models. arXiv (2024). https://doi.org/10.48550/arXiv.2404.01589

[3] Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, Renqian Luo, Scott Mayer McKinney, Robert Osazuwa Ness, Hoifung Poon, Tao Qin, Naoto Usuyama, Chris White, Eric Horvitz: Can Generalist Foundation Models Outcompete Special-Purpose Tuning? Case Study in Medicine. arXiv (2023). https://doi.org/10.48550/arXiv.2311.16452

[4] Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Nenad Tomasev, Yun Liu, Renee Wong, Christopher Semturs, S. Sara Mahdavi, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, Vivek Natarajan: Towards Expert-Level Medical Question Answering with Large Language Models. arXiv (2023). https://doi.org/10.48550/arXiv.2305.09617

[5] OpenAI: "GPT-4-o Mini: Advancing Cost-Efficient Intelligence." OpenAI, August 2024, https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

[6] American Joint Committee on Cancer: AJCC Cancer Staging Manual (8th ed.). Springer, 2017.

[7] Sajjad Abedian, Evan T. Sholle, Prakash M. Adekkanattu, Marika M. Cusick, Stephanie E. Weiner, Jonathan E. Shoag, Jim C. Hu, and Thomas R. Campion Jr: *Automated Extraction of Tumor Staging and Diagnosis Information From Surgical Pathology Reports.* PubMed (2021). https://pubmed.ncbi.nlm.nih.gov/34694896/

[8] Hidetoshi Matsuo, Mizuho Nishio, Takaaki Matsunaga, Koji Fujimoto, Takamichi Murakami: *Exploring Multilingual Large Language Models for Enhanced TNM Classifi-*

cation of Radiology Report in Lung Cancer Staging. 2024. https://arxiv.org/pdf/2406.06591

[9] Jan Clusmann, Fiona R. Kolbinger, Hannah Sophie Muti, Zunamys I. Carrero, Jan-Niklas Eckardt, Narmin Ghaffari Laleh, Chiara Maria Lavinia Löffler, Sophie-Caroline Schwarzkopf, Michaela Unger, Gregory P. Veldhuizen, Sophia J. Wagner, and Jakob Nikolas Kather: *The future landscape of large language models in medicine.* Nature, 2023. https://www.nature.com/articles/s41591-023-02573-3

[10] Hyung Jun Park, Namu Park, Jang Ho Lee, Myeong Geun Choi, Jin-Sook Ryu, Min Song, and Chang-Min Choi: *Automated extraction of information of lung cancer staging from unstructured reports of PET-CT interpretation: natural language processing with deep-learning.* BMC Medical Informatics and Decision Making, 2022. https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-022-01975-7

[11] Zabir Al Nazi and Wei Peng: *Large Language Models in Healthcare and Medical Domain: A Review.* MDPI, 2023. https://www.mdpi.com/2227-9709/11/3/57

[12] DeepSeek-AI: *DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning.* https://arxiv.org/pdf/2501.12948, 2023.

[13] Llama Team, AI @ Meta: *The Llama 3 Herd of Models.* https://arxiv.org/pdf/2407.21783, 2024.