Universiteit Utrecht

Capgemini

Exploring AI-Assisted Triage: Comparing Classical Machine Learning and GPT-4o for Clinical Decision Making

44 ECTS Artificial Intelligence Master Thesis

Supervisor & Reviewer:
Dr. ir. J.M. (Jan) Broersen

Supervisor Capgemini:
L. (Linda) van de Krol MSc

Second Reviewer:
Dr. J. (Joris) Hulstijn

Author:
Emma J. Braam
6412300

April 13, 2025

Abstract

Accurate and efficient triage is essential to optimize the allocation of limited medical resources in the emergency department (ED). Nevertheless, triage decisions are inherently hard, requiring rapid but justifiable judgments under time constraints, incomplete patient data, and uncertainty. In such circumstances, large language models (LLMs) like GPT-4o introduce a new potential by leveraging their advanced capabilities. However, concerns about the transparency and reliability of LLMs persist, particularly in high-stakes settings such as triage.

One promising direction is to constrain LLMs to use structured rule-based reasoning, ideally in a way that reflects how triagists reason. In triage, clinicians often rely on default reasoning, which means that they rely on general assumptions unless contradicted by specific evidence. The BOID framework (Beliefs, Obligations, Intentions, Desires) builds on this by representing different mental attitudes and resolving conflicts between them through structured prioritization. Integrating BOID into LLMs could potentially allow users to trace which default rules from which mental attitudes were prioritized in particular triage decisions.

While this thesis does not implement a full BOID-LLM system, we take a first step by focusing on the O-component. We treat triage decisions from our real-world Korean Triage and Acuity Scale (KTAS) data as obligation-driven and explore whether GPT-4o can simulate obligation-based reasoning, without the support of retrieval mechanisms. We compare GPT-4o's performance using multiple prompt engineering techniques against classical machine learning (ML) models including Decision Trees (DT), Random Forests (RF), and eXtreme Gradient Boosting (XGBoost). We further extract triage decision rules from GPT-4o and compare them with feature importance insights from applying SHapley Additive exPlanations (SHAP) on our best performing classical ML model. Finally, we reflect on ethical concerns, including reliability, fairness, transparency, and data privacy.

Our results showed that RFs slightly outperformed DTs and XGBoost. Among prompt engineering techniques, integrating SHAP-derived results from our best performing classical ML model in the prompt improved GPT-4o's performance. However, even its best configuration (Weighted Kappa = 0.6186) fell short of all classical ML models. Applying this to real-world triage would mean that a significant proportion of patients would still be misprioritized. Furthermore, GPT-4o seemed to rely on general medical knowledge rather than explicit instructions in the prompt. In addition, GPT-4o showed inconsistencies in its extracted triage rules and showed a gap between how it claims to reason and what it actually predicts. These results raise ethical concerns when using LLMs like GPT-4o for high-stakes clinical tasks such as triage decision making and emphasize the need for a hybrid BOID-LLM system that combines the explainability and structure of default logic with LLMs to support safe and reliable decision making in triage.

Acknowledgements

Over the past eight months, I have been fully immersed in writing my master's thesis for the AI master's program. However, I would not be standing here today without the support of several people.

First and foremost, I would like to thank my supervisor at Utrecht University, Jan. Especially in the early stages of this project, when the direction of my thesis was still uncertain, we had frequent, often weekly, meetings. I am very grateful for his guidance, direction, and for listening to my thoughts throughout the entire process. I always felt comfortable asking questions, and that openness was incredibly valuable to me.

I would also like to thank Linda, my supervisor at Capgemini. I truly appreciated our moments of sparring and discussing the state of my work with her. Furthermore, I am also thankful for the feedback provided by my second reader, Joris. His input helped shape the final version of this thesis.

Last but not least, I want to thank my family and friends for their constant support. This thesis has taken many long hours and brought its fair share of stress, and I am deeply grateful to have had you by my side.

Contents

1 Introduction

In this section, we present an overview of the main features of our research. We start with background information on large language models (LLMs) and their possible applications in healthcare. The Problem Statement specifies the challenges in using LLMs in this field and briefly mentions the role of default reasoning in triage decision making. The Research Objectives specify the two-fold concentration of the study: (1) a data-driven approach, where we evaluate the performance of classical machine learning (ML) models Decision Trees (DT), Random Forests (RF), and eXtreme Gradient Boosting (XGBoost) in predicting Korean Triage and Acuity Scale (KTAS) levels and (2) a prompt engineering approach, where we examine the ability of GPT-4o to generate triage rules and to predict triage levels based on various prompting techniques. We will present our main research questions along with our subquestions that guide our investigation. The section concludes with an overview of the research structure.

1.1 Background

The development of LLMs like BERT [32] and GPT [171] has driven significant advancements in artificial intelligence (AI) in recent years. From text creation and natural language comprehension to complex decision making challenges, these models demonstrate impressive skills [83, 87, 146]. In healthcare, their potential extends in various applications, including medical education, text-based diagnostics, genetic analysis, medical summary generation, patient management, and clinical decision support [12, 93, 137, 168]. With their ability to process large volumes of text and generate coherent, and contextually appropriate responses, LLMs offer exciting promises to improve decision making in critical clinical settings.

A particularly high-stakes decision making scenario in healthcare is triage, which involves prioritizing patients based on the urgency of their medical conditions. Triage systems are widely implemented to optimize the emergency department (ED) workflow, allocate resources efficiently, and ensure timely medical intervention [72, 119]. However, this process often presents ethical dilemmas, such as respect for patient autonomy, nonmaleficence ("do no harm"), and justice (fair allocation of decisions) [1]. In addition, triage decision are inherently hard, as clinicians have to make rapid but justifiable decisions under time constraints, incomplete patient data, and uncertainty. In such settings, decision support tools that leverage AI could enhance accuracy and consistency in triage classifications.

1.2 Problem Statement

The complexity in triage decisions makes it compelling to explore how LLMs can support triage decisions. There are, nevertheless, severe uncertainties when using LLMs in such high-stakes environments. An important example is that LLMs are black boxes. This makes their decision making process impenetrable and impossible to comprehend [130]. In emergency medicine, where accountability and transparency are critical, the use of an AI system lacking explainability is undesirable.

In addition, LLMs suffer from hallucinations and nonsensical reasoning [36, 170], raising concerns about their reliability in life-critical contexts. Furthermore, studies investigating the performance of LLMs in triage have shown that they can misdiagnose or overestimate severity, leading to incorrect prioritization of patients [37, 95]. Also, LLMs cannot be expected to uphold all ethical principles and values that guide medical providers, as they lack intrinsic understanding of human duties and needs.

One promising direction to address these uncertainties is by integrating structured rule-based reasoning systems into LLMs, offering the potential to constrain LLMs to reason within predefined reasoning frameworks. Ideally, we would constrain LLMs to reason within a structured rule-based reasoning system that reflects how clinicians reason when making triage decisions.

In triage, clinicians often depend on default reasoning. This means that they make assumptions based on common medical patterns unless they encounter evidence that contradicts those assumptions. For example, a patient who arrives with chest pain might be classified as urgent by default, unless additional information indicates this patient can safely wait. This kind of reasoning is the basis of the BOID framework (Beliefs, Obligations, Intentions, Desires) [15, 16], which incorporates default logic and structured prioritization to manage uncertainty and conflicting mental attitudes in decision making.

The BOID represents these different mental attitudes, and in triage, these can be interpreted in the following way. *Beliefs* are default hypotheses that clinicians hold based on typical medical presentations, which are revised as more patient information becomes available, *obligations* are clinical guidelines, standardized protocols, and ethical responsibilities that healthcare providers have to follow when deciding how to prioritize patient care, and *desires* are the higher-level objectives of clinicians, primarily optimal patient outcomes and efficient use of resources. *Intentions* are unique because they represent either past triage decisions or a planned course of action, shaped by a balance of beliefs, obligations, and desires. These components regulate decision making in situations of uncertainty where multiple conflicting mental attitudes must be considered. By combining them into a hybrid system, BOID leverages the potential to extract what default rules are being used and how different mental attitudes are prioritized for a specific triage decision. This could result in a more transparent, trustworthy, and manageable AI.

While the complete integration of the BOID into LLMs is a long-term goal, this thesis takes a first step by focusing on the obligations (O) component. We treat the triage decisions from our real-world triage data as obligation-driven and

explore whether GPT-4o can simulate obligation-based reasoning, without relying on information retrieval from the data. To evaluate GPT-4o's capabilities, we tested multiple prompting techniques and compared their performances against classical machine learning (ML) models including Decision Trees (DT), Random Forests (RF), and eXtreme Gradient Boosting (XGBoost).

1.3 Research Objectives

To investigate the predictive capabilities of classical ML models and GPT-4o in classifying triage levels, we employ an open-source Korean Triage and Acuity Scale (KTAS) dataset from Kaggle [182]. KTAS is a five-level triage system widely used in EDs in South Korea. It is designed to categorize incoming patients into the most suitable KTAS label based on the severity of the patient's medical condition. Patients are given a level of urgency ranging from 1, the most urgent patients, where patients have to be treated immediately, to 5, the least urgent patients, where patients can safely wait. The dataset consists of 16 variables, including demographic and clinical data, and the chief complaint of the patient. Furthermore, for each patient, the dataset includes two independent triage assignments: one by registered nurses and another by triage experts. More details on this and a particular classification example are provided in Section 3.2.

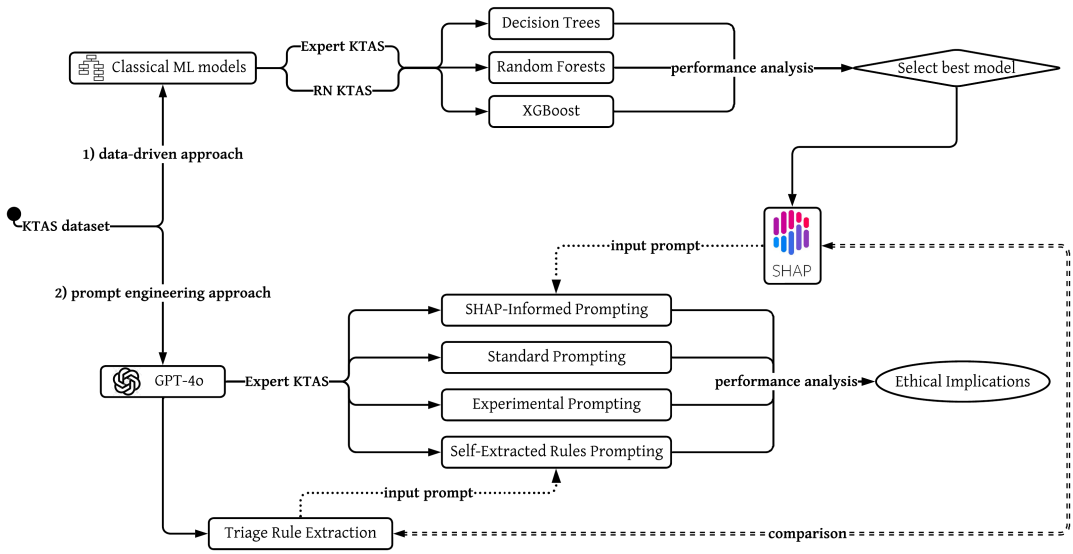


Figure 1.1: Schematic Overview of the Research

As illustrated in Figure 1.1, this thesis pursues two main research directions. First, a data-driven approach, in which we evaluate the performance of classical ML models DTs, RFs, and XGBoost in predicting KTAS triage levels, and second, a prompt engineering approach, in which we examine the ability of GPT-4o to generate explicit triage rules and to predict triage levels based on varied prompting techniques.

In the first direction, the data driven approach, we evaluate the performance of DTs, RFs, and XGBoost in predicting KTAS levels assigned by either registered nurses or triage experts. This unique feature in our dataset allows an exploration of how different clinical perspectives inform triaging decisions. On the one hand, registered nurses frequently use heuristic judgments and pattern recognition, developed through training and practical experience, to rapidly assess patient conditions in high-pressure environments. On the other hand, triage experts, which we will consider to assign the gold standard KTAS labels, may possess more specialized knowledge when encountering familiar clinical scenarios, allowing them to make swift decisions based on prior experience and recognized patterns. By analysing the differences between KTAS assignments from registered nurses and experts, we aim to quantify how well our classical ML models compare to these triage decision makers. To evaluate this, we used key performance metrics to assess the effectiveness of the models. An overview of the design of this specific part of the research can be found in Figure 1.2. This figure will also be helpful for understanding the methodology used, as detailed in Section 3.5. Additionally, we also apply SHAP analysis on the best performing classical ML model to identify the most influential features in this triage classification task.

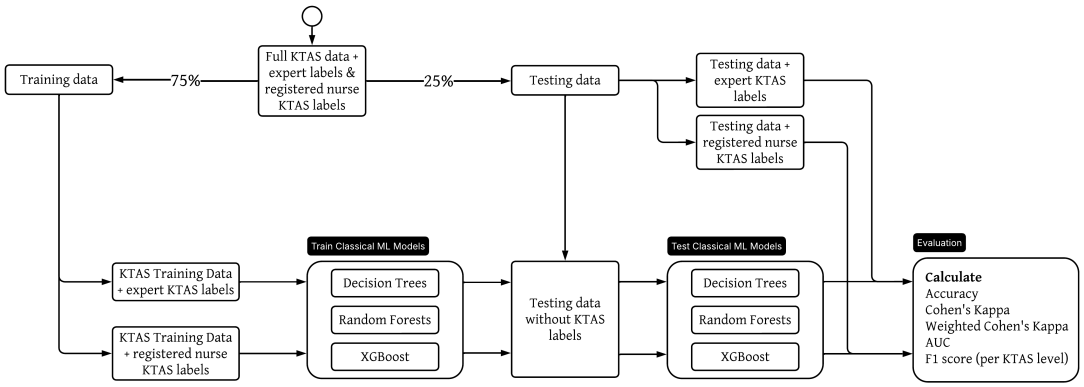


Figure 1.2: Modeling Design Classical ML Approach for KTAS Prediction

The second direction investigates the ability of GPT-4o to predict KTAS expert triage levels. We start by analysing the rules extracted from GPT-4o when prompted to generate KTAS triage decision rules, without providing any KTAS-specific examples. However, we do explain to the model how our KTAS data is built, providing all possible variables and corresponding values. The resulting rules are then compared to the SHAP-extracted feature importance results from applying SHAP on the best performing ML model. The aim is to assess to what extent the inferred triage rules of GPT-4o align with the gold standard expert KTAS labels. Also, while the specific sources used to train GPT-4o are unknown [25], we investigate whether GPT-4o, to some extent, can provide deeper insights into specific combinations of patient values corresponding to particular KTAS levels compared to SHAP's ranking of feature importances. Next, we evaluate GPT-4o's ability to predict KTAS levels using a variety of prompting techniques, including standard prompts, SHAP-informed prompts, various experimental prompts, and

a prompt incorporating GPT-4o's own extracted triage rules. This specific part of the research, in which we let GPT-4o predict KTAS levels using various prompting techniques, can be found in Figure 1.3. The modeling design will be further detailed and explained in Section 3.6. Finally, we discuss the ethical implications of using GPT-4o in triage systems, addressing concerns that focus on reliability, fairness, transparency, and data privacy.

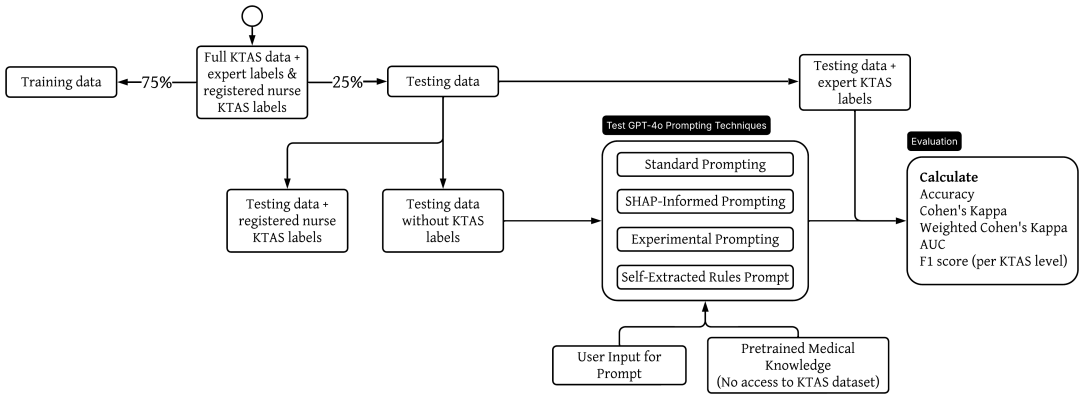


Figure 1.3: Modeling Design GPT-4o Prompting Approach for KTAS Prediction

Together, these directions aim to answer the following main research questions.

> **To what extent can GPT-4o predict KTAS triage levels, and how do different prompt engineering techniques affect its performance compared to classical ML models?**

To address this, we investigate the following four subquestions.

SRQ1. How do classical ML models, specifically DTs, RFs, and XGBoost, compare in their ability to predict registered nurse-assigned and expert-assigned KTAS levels?

We hypothesize that the XGBoost model will outperform both DT and RF in predicting KTAS levels for both prediction tasks. This is expected because XGBoost, in contrast to DT and RF, is able to handle imbalanced datasets and capture more complex feature interactions [78]. We further hypothesize that predicting expert KTAS would result in better performance than predicting registered nurse KTAS, as we expect that decisions made by experts are more consistent than decisions made by registered nurses because the former group has longer clinical experience [135].

SRQ2. What are the most influential variables affecting KTAS predictions for the best performing classical ML model?

We hypothesize that important influential variables will largely correspond to common medical knowledge. For example, we expect that complaints such as cardiac arrest and chest pain will indicate high urgency levels, while complaints such as medication refill, procedures, and follow-ups will indicate

lower urgency levels. Furthermore, we hypothesize that an unresponsive mental status will significantly influence higher urgency levels, whereas being alert will correspond to lower urgency levels. Additionally, we hypothesize that low saturation values and arrival by ambulance will indicate higher urgency levels, while high saturation values and arrival by private vehicle or by foot will indicate lower urgency levels.

SRQ3. To what extent can GPT-4o generate clinically meaningful triage decision rules, and how do these compare to the results of applying SHAP on classical ML models?

We hypothesize that GPT-4o will generate triage rules that partially align with the feature importance rankings derived from SHAP analysis of classical ML models, particularly for clear high- and low-urgency indicators. While GPT-4o is not given access to our KTAS dataset or exposed to any examples, it is provided with a structured overview of the dataset variables, their possible values, and their meanings. Based on this metadata and its general medical knowledge from pretraining, we expect GPT-4o to associate features such as unresponsiveness, cardiac arrest, and abnormal vital signs with high urgency, and features like medication refill, follow-up visits, or stable vitals with low urgency.

SRQ4. How does using various prompting techniques (namely, standard prompting, SHAP-informed prompting, experimental prompting, and prompting with GPT-4o's own extracted triage rules) influence the performance of GPT-4o to classify KTAS levels, and how does its performance compare to classical ML models?

We hypothesize that the standard prompting techniques will improve GPT-4o's performance over zero-shot prompting, our baseline prompting method, which does not provide any additional context besides the patient data. This is expected because research has shown that these standard prompting techniques enhance the reasoning capabilities of LLMs [70, 84, 149]. Furthermore, we hypothesize that SHAP-informed prompting, which integrates key variables derived from model interpretability, will also lead to improved performance by guiding the model's attention to clinically relevant features [105]. For experimental prompting, we include emotionally framed prompts, contextual constraints (e.g., variable limits), and contradictory instructions. We hypothesize that incorporating emotional framing may slightly improve GPT-4o's performance [80], while we expect GPT-4o's performance to decrease when experimenting with cognitive noise or misalignment with medical logic. Also, we hypothesize that prompting with GPT-4o's own derived default rules will at least achieve GPT-4o's performance when using our baseline prompting. We further hypothesize that, under optimized prompting, GPT-4o may outperform classical ML models in KTAS classification due to its broad medical knowledge and flexible reasoning capabilities, which go beyond the pattern-based limitations of our ML models trained solely on structured data.

This thesis is relevant to the fields of explainable AI and clinical decision support

systems. With these subquestions, we aim to contribute to the investigation of the performance and ethical utility of using LLMs in triage. We explore the potential of obligation-based reasoning through LLM prompting, offering an initial step towards integrating default logic frameworks like BOID with LLMs.

1.4 Thesis Organization

Section 2 presents the necessary background information, starting with an overview of triage systems and the KTAS. Then it delves into LLMs, in particular GPT-4o, and their capabilities and limitations. This is followed by background on enhancing their reasoning abilities through prompt engineering and integration of external tools. Additionally, the classical ML models DTs, RFs, and XGBoost will be introduced, along with a method for interpretability using SHAP analysis. Finally, the BOID framework is presented, highlighting its relevance to triage decision making.

Chapter 3 explains the methodology used for KTAS level prediction and triage decision making analysis. It covers data preprocessing and class imbalance mitigation. It also outlines the methodologies used for DTs, RFs, and XGBoost, as well as for the extraction of triage rules from GPT-4o and the various prompting approaches to test GPT-4o's performance in KTAS classification.

Section 4 presents the results, beginning with the performances of DT, RF, and XGBoost. This will be followed by the results of applying SHAP values on the best performing classical ML model to determine the most influential variables in triage decision making. Then, the section delves into the results obtained from extracting triage rules from GPT-4o and the results from applying the various different prompting techniques on GPT-4o.

Section 5 provides a detailed discussion of the results and compares our findings with existing research. It also explores the potential integration of the BOID framework into LLMs as a possible approach to addressing the current challenges of using LLMs such as GPT-4o in triage. Additionally, this section addresses the ethical implications of LLM-driven triage systems and discusses how the results from our prompt engineering techniques provide insight into these implications. Lastly, we point out the limitations of the research.

Section 6 concludes the thesis by systematically addressing our research subquestions before synthesizing the findings to formulate a response to our main research questions.

2 Background

This section provides background on triage systems, LLMs, and classical ML models, setting the foundation for the research in this thesis. We begin with an overview of triage systems, including KTAS, followed by an exploration of LLMs with, in particular, GPT-4o, and methods for improving their reasoning capabilities. The section also examines classical ML models DTs, RF, and XGBoost, and SHAP values as an interpretability method for these models. Finally, we introduce the BOID framework and illustrate its relevance to triage decision making. This final section marks a conceptual shift, moving from data-driven learning to structured decision making grounded in default logic.

2.1 Triage Systems and KTAS

Triage refers to the process of assessing the severity of the illness or injury of a patient shortly after arrival at the ED, which serves as the critical interface between emergency medical services and medical care [22]. It ensures that those in need of urgent care receive timely treatment while optimizing resource allocation for less critical cases. Triage systems vary from simple, experience-based classifications (using the 'best guess') [24] to structured three-level systems (emergent, urgent, non-urgent) [73], four-level systems [117] and five-level systems. However, five-level triage systems provide better validity and reliability in emergency care [147]. Some widely used five-level triage scales include the Australasian Triage Scale (ATS), the Manchester Triage System (MTS), the Emergency Severity Index (ESI), and the Canadian Triage and Acuity Scale (CTAS) [22].

In this study, we utilize the Korean Triage and Acuity Scale (KTAS), a five-level triage scale implemented in 2012 and derived from CTAS [118]. The KTAS process, as visualized in Figure 2.1, is designed to help emergency nurses assign a triage level after conducting an initial critical assessment [63, 101, 131]. It assigns patients a priority level based on clinical urgency, determining the maximum allowable wait time before evaluation and ensuring that critical cases receive timely care [118]. The KTAS has also been shown to reduce length of stay and mortality in EDs [71].

Despite these systems, the quality of decisions made is subject to substantial variation and have been shown to be affected by the personal characteristics of the healthcare provider, such as their experience level and vulnerability to fatigue during triaging [35, 175]. Furthermore, misclassification, over-triage, and under-triage persist because of the inherent complexity and uncertainty of triage [21]. To address such issues, LLMs are becoming increasingly popular to enhance triage precision and support clinical decision making.

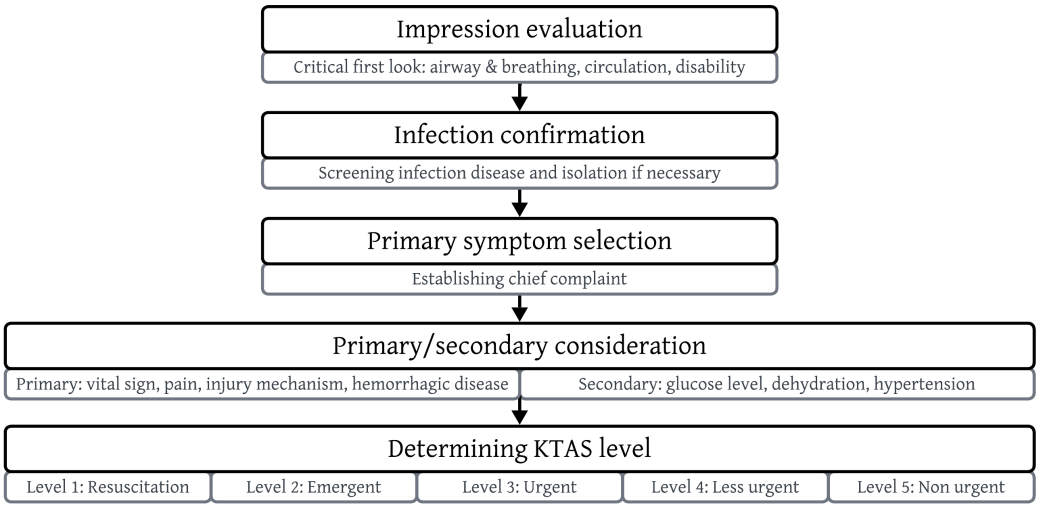


Figure 2.1: Process of the Korean Triage and Acuity Scale (KTAS) [101]

2.2 Large Language Models

LLMs like Bidirectional Encoder Representations from Transformers (BERT) [32] and Generative Pre-trained Transformers (GPT) [171], have revolutionized the field of natural language processing (NLP). The models are transformer-based [67], originally introduced by Vaswani et al. [151] in "Attention Is All You Need." The architecture incorporates a multi-head self-attention mechanism, which allows models to process multiple input segments in parallel. With the ability to handle long-range dependencies in text and encode complex contextual relationships, transformers have been found to possess superior capabilities in generating fluent and coherent text. This enabled LLMs to show remarkable performance on tasks ranging from text analysis to decision making support [13, 83, 87, 179, 180].

In healthcare, LLMs have the potential to transform numerous aspects, such as diagnostics and clinical decision support [43, 144, 169]. By processing extensive medical records and intricate clinical information, LLMs can assist users in producing meaningful insights to support decision making and patient management [52, 82, 138]. In addition, LLMs can integrate clinical knowledge with contextual reasoning, which can enable them to replicate the thought processes that healthcare providers use during triage.

2.2.1 Generative Pre-trained Transformer 4o

One of the greatest breakthroughs achieved in NLP was the GPT [51]. The GPT series was introduced by Radford et al. [121], and with each subsequent release, scale and performance increased. Its decoder-only architecture allows them to generate coherent and contextually relevant text in an autoregressive manner, making them particularly well suited for tasks such as text completion, content generation, and conversational AI. Furthermore, since they have been pre-trained on huge quantities of unlabeled text data using self-supervised learning, they can generate

text without supervision [122]. Over time, GPT-based models have found significant applications in various industries, including finance [76], customer service [128], and healthcare [88], where their ability to process and generate language efficiently has proven to be highly valuable. Furthermore, the widespread adoption of GPT-based models, particularly through applications like ChatGPT, has made AI-driven interactions part of daily life.

The latest iteration of the GPT-series, GPT-4o, was introduced in May 2024 and, at the time of writing, is only accessible through a paid subscription for ChatGPT Plus users or a pay-per-use model when used with an API key. It is built as a multimodal multilingual transformer [160, 161], which implies that it can combine and handle various data types, such as text, images, audio, or video. Compared to older models, GPT-4o offers massive efficiency improvements without compromising on accuracy and performance levels. These advancements offer faster response times and reduced costs compared to GPT-4, while maintaining high accuracy and performance, surpassing the improvements seen in GPT-3 and GPT-3.5 [59].

For those interested in learning more about GPT models, Qiu et al. [120] provide a detailed description of their development, architectural innovations, and practical applications.

2.3 Enhancing Reasoning Abilities in LLMs

While LLMs are promising, several studies indicate the need for further research before using them in sensitive areas such as triage. For example, a comparison of various LLMs such as ChatGPT revealed that while GPT-4-based ChatGPT was nearly as good as the nonexpert doctors' decisions, it had still not achieved the accuracy of professional doctors [95]. In particular, LLMs tended to over-triage, where the condition of a patient is improperly assessed to be more urgent than it actually is, possibly leading to unnecessary prioritization and burden on the emergency services. In other words, this study suggests that LLMs tend to be risk averse. Conversely, nonspecialists often under-triage, assessing the conditions of a patient to be less serious than they actually are, thereby postponing important treatment and influencing outcomes for patients. This finding is further supported by the research of Haim et al. [49], who compared the Emergency Severity Index (ESI) score of GPT-4 with human evaluators.

In addition, a study that assessed the diagnostic accuracy of ChatGPT using CTAS concluded that the model was accurate in just 47.5% of the cases and had both high under-triage and over-triage rates [37]. An examination of the performance of LLM in real-world ED chest pain triage in real life also demonstrated inconsistency in the performance of the model with regard to accuracy and speed when compared to human experts [96]. These findings indicate the limitations of current LLMs in delivering accurate and reliable triage results, emphasizing the need to improve their reasoning capabilities in critical medical scenarios.

To address the limitations of LLMs, efforts have been made to improve their performance. Generally speaking, this research falls into (1) prompt engineering and (2) tool-augmented approaches.

2.3.1 Prompt Engineering Strategies

The first approach is prompt engineering, also referred to as prompt optimization. In NLP, a *prompt* refers to the input or instruction given to an AI to specify the task it should perform [122]. A prompt typically comprises several components that guide the model's response: a clear instruction specifying the task for the model, additional information that helps steer the model toward a more accurate or relevant response, a specific question or data for which a response is wanted, and an indication of the desired structure of the response of the model [47]. Specifically designing these inputs to guide the AI to achieve the most optimal and efficient outputs, without altering the model's parameters, is known as *prompt engineering* [85, 159]. Prompt engineering has been intensively investigated in medicine as a method to improve the performance of NLP applications in the medical sector [154].

The simplest form of prompting an LLM is *zero-shot prompting*, in which the model is expected to perform a task without any prior examples or additional context [84]. With *few-shot prompting*, on the other hand, the model is provided with a few examples of the desired outcome [84]. These are used to guide the pre-trained model to perform particular tasks [86].

Several methods have been developed for optimizing these prompts [132]. One of them is *Chain-of-Thought (CoT) prompting*. The CoT prompting method has been shown to successfully improve LLM performance on complex reasoning tasks by engaging their reasoning ability [155, 157, 158, 177, 178]. CoT prompting can be categorized into two strategies: explicit prompting, where the problem is decomposed directly within the prompt, and implicit prompting, where the general reasoning cue "let's think step-by-step," is added to the end of the prompt to encourage sequential reasoning, resulting in a more transparent reasoning process [85, 149]. Furthermore, *role prompting*, where the LLM is assigned a specific persona or identity, has been shown to act as an implicit CoT trigger, improving reasoning performance by immersing the model in a task-specific context [70].

Some more examples are *least-to-most prompting* [181], which simplifies complex problems by breaking them into smaller components, and OPRO, where LLMs iteratively refine their solutions, leading to substantial improvements in task performance, particularly in the area of prompt optimization [167].

In Section 5.2.2, we examine previous work in which prompt engineering methods are particularly used in triage tasks. We will also compare these with our research.

2.3.2 Tool-Augmentation Strategies

The second approach to enhance the reasoning of LLMs involves integrating them with external tools. These are resources the model can use during a task, but were not part of its original training, such as knowledge bases or reasoning modules. This field has produced several notable frameworks that combine LLMs with symbolic reasoning systems.

Integrating LLMs with external tools or symbolic systems enhances their reasoning capabilities by addressing their limitations in logic and decision making.

Approaches such as LogicLM [115], which involves the combination of LLMs with symbolic solvers for self-refinement, and LLM-ARC [64], which applies a neuro-symbolic Actor-Critic framework for logical program refinement, exemplify this method. Similarly, LINC Olausson et al. [110] uses LLMs as semantic parsers to parse natural language to first-order logic for theorem proving, and Argumentative LLMs Freedman et al. [38] produce structured arguments and rebuttals that facilitate transparency of complex decision-making tasks. This latter example is particularly interesting when it comes to non-monotonic reasoning, such as default reasoning.

In healthcare, tools such as In-Context Padding (ICP) [165] integrate clinical knowledge to enhance LLM-based diagnostic reasoning. GeneGPT [62] integrates LLMs with biomedical databases to improve question-answering accuracy. For triage-specific applications, TriageAgent [89] incorporates confidence scoring and early-stopping mechanisms, outperforming existing LLM-based triage frameworks.

The ultimate idea of creating a BOID-LLM, of which we will investigate an initial step, would use the second approach, since this integration lets LLMs use the BOID architecture as an external tool, providing a structured mechanism to handle competing priorities during triage. The BOID framework will be introduced further in Section 2.5.

Rather than fully integrating BOID into GPT-4o as an external reasoning module, we take a first step toward that vision by focusing on the obligation-component of the BOID framework. Therefore, this work remains situated within the first approach, using prompt engineering alone to investigate whether GPT-4o can generate structured triage rules, by comparing them with SHAP analysis on the best performing classical ML model. We also evaluate its prediction performance under different prompting techniques and compare these results against the classical ML models.

2.4 Classical ML Models

While LLMs offer a novel approach to automated decision making in triage, classical ML models such as DTs, RFs, and XGBoost remain widely used due to their structured decision making processes, interpretability, and strong predictive performance in healthcare applications [18, 30, 34, 42, 68]. These models provide a valuable comparison point for evaluating LLM-driven triage through a direct comparison of predictive performance across multiple metrics, including class-specific performance.

2.4.1 Decision Trees

DTs represent one of the most popular methods of classification and regression problems in supervised ML [129]. They provide a comprehensible and interpretable framework for decisions within a huge amount of complex data. DTs create partitions of the input space into regions defined by a set of hierarchical rules. Each internal node tests a feature, while every leaf provides a classification or an esti-

mated value. Their comprehensibility makes DTs particularly suitable for domains that require transparency, such as healthcare.

DTs start at the root and divide the dataset into subsets based on the values of their features. The choice of feature and split point is determined by criteria such as Information Gain or Gini Impurity [27].

- **Information Gain** measures the entropy reduction achieved by partitioning a dataset based on a particular feature. Entropy measures the level of disorder or impurity within a dataset. The values of entropy lie between zero and one, with zero meaning that all samples in a dataset S belong to a single class and one meaning that all samples belong to different classes. To construct an optimal DT, the feature a_i that minimizes entropy (or maximizes Information Gain) is selected for splitting. Information Gain is calculated as the difference in entropy before and after a split on a given feature:

$$InformationGain = Entropy_{parent} - Entropy_{children}, \tag{2.1}$$

  with

$$Entropy_x = -\sum_{i=1}^{n} p_i log_r(p_i) \tag{2.2}$$

  The feature with the highest Information Gain provides the most effective split, as it best separates the training data according to the target classification.

- **Gini Impurity** also is an impurity-based criterion. It quantifies how often a randomly chosen data point would be misclassified if it were labeled according to the class distribution within the dataset. In essence, it measures the divergence from a perfectly pure class distribution, using the following formula:

$$Gain = 1 - \sum_{i-1}^{j} P(i)^2 \tag{2.3}$$

  Like entropy, the Gini Impurity equals zero for a pure set (i.e., when all data points belong to a single class).

These methods aim to create subsets that are as pure as possible. The splitting process is applied recursively to each subset, resulting in a hierarchical tree structure.

Some of the main reasons why DTs are popular include simplicity, interpretability, and flexibility [129]. Furthermore, DTs intrinsically provide feature importance that can be helpful to understand what variables are influencing the predictions. Also, they are capable of handling datasets that may have missing values or errors.

However, they do have some key weaknesses, such as overfitting to the underlying data they were trained on, being sensitive to slight changes in the underlying dataset, and having biases toward features with a greater number of levels. Moreover, DTs rely on a greedy splitting algorithm that may not always make optimal choices. Their performance may be affected by noisy data or outliers.

2.4.2 Random Forests

RFs, therefore called a bagging method, resolve these issues by creating multiple DTs on random subsets of the dataset, with the final prediction being the class most commonly predicted by the individual trees [14]. By generating multiple trees, RFs reduce overfitting and provide more stable predictions compared to single DTs.

Compared to Decision Trees, RFs are less interpretable due to their nature as an ensemble. But this also makes them less prone to overfitting and more robust against outliers. Furthermore, while RFs need more computational power, they generally provide greater accuracy than DTs, especially for large multivariate datasets with many features [94].

2.4.3 Extreme Gradient Boosting

Like RFs, XGBoost is an ensemble method as well. But instead of voting over the most predicted outcome, it builds trees sequentially, trying to gradually correct the mistakes of the previous trees so that optimal performance is achieved [20]. The process is called gradient boosting, which minimizes a loss function using the gradient to make better predictions at each step. In this way, if the model struggles to classify certain samples, particularly from minority classes, subsequent trees give special priority to these difficult cases by increasing their influence in the learning process. By doing so, the model's attention is drawn towards more difficult-to-predict instances, helping to mitigate class imbalance [78].

One of XGBoost's key innovations is the introduction of a regularized learning objective, balancing predictive accuracy and model complexity. This is achieved by minimizing the following objective:

$$L(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k), \tag{2.4}$$

where

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2 \tag{2.5}$$

Here, $l(\hat{y}_i, y_i)$ measures the difference between the prediction $\hat{y}_i$ and the target y_i, while $\Omega(f_k)$ penalizes model complexity through the number of tree leaves (T) and the leaf weights (w). This regularization reduces overfitting, which is a common issue in both DTs and RFs.

XGBoost models are popular since they excel in performance, efficiency, and versatility, handling various data types while preventing overfitting through built-in regularization. Its parallel processing and memory optimizations enable fast

training. However, applying XGBoost requires careful hyperparameter tuning and its ensemble structure makes interpretation challenging compared to simpler models [164].

2.4.4 Interpretability through SHAP Analysis

The classical ML models used in this study, DTs, RFs, and XGBoost, are powerful but vary in interpretability. While DTs are relatively understandable because of their hierarchical structure, RFs and XGBoost are complex ensemble methods whose impact of individual features towards the prediction cannot be directly determined. However, not having a clear understanding why the model predicts a particular KTAS level for a patient raises ethical questions about using its predictions because triage decisions directly affect patient outcomes. Therefore, their lack of interpretability is a barrier to AI implementation in high-stake clinical settings.

A number of interpretability techniques are developed to offer explanations of ML models. Partial Dependence Plots (PDPs) plot the marginal effect of one or two features on the predicted target by averaging over all instances [39]. Individual Conditional Expectation (ICE) extends PDPs by plotting how a feature affects individual predictions, thus giving a more nuanced view of its influence as opposed to averaging over the dataset [44]. Permutation Feature Importance quantifies the global importance of a feature by randomly shuffling its values and measuring the corresponding change in model performance [14]. Local Interpretable Model-Agnostic Explanations (LIME) generates localized, sparse explanations by approximating a black-box model with a simpler, interpretable surrogate model [127].

SHapley Additive exPlanations (SHAP), rooted in cooperative game theory, provides a strong and theoretically robust interpretability approach [90]. Whereas LIME focuses on local explanation of particular instances, SHAP has both local (instance-level) and global (dataset-level) interpretability. SHAP values are used to interpret the output of the ML model by approximating the marginal contribution of every feature to the predictions of the model. At its core, SHAP distributes a model's output across its features based on their individual contributions. This ensures consistency and accuracy in attributing importance [166], which is particularly crucial in high-stakes applications like triage. Additionally, SHAP offers a variant specifically designed for tree-based models, TreeSHAP [91]. As DTs, RFs, and XGBoost are examples of tree-based models, it is especially suitable for our research.

The mathematical foundation of SHAP values is as follows. Consider a predictive model as a cooperative game, where the features of the dataset represent the players. The SHAP value for a given feature i is calculated by averaging its marginal contribution across all possible subsets of features S that do not include i. Formally, the SHAP value ϕ_i for feature i is expressed as:

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \cdot \Big(v(S \cup \{i\}) - v(S) \Big), \tag{2.6}$$

where N is the set of all features and S is a subset of features N excluding i. $v_x(S)$ represents the model's predictive value when only the features in set S that are marginalized over features that are not included in set S:

$$v_x(S) = \int \hat{f}(x_1, \ldots, x_p) d\mathbb{P}_{x \notin S} - \mathbb{E}_X[\hat{f}(X)] \tag{2.7}$$

Furthermore, $\frac{|S|!(|N|-|S|-1)!}{|N|!}$ is a combinatorial weight ensuring fair distribution of contributions.

This formula satisfies several desirable properties, such as efficiency, symmetry, and linearity, making it particularly suitable for feature attribution. When applied to classical ML models, SHAP values can provide detailed insights into the importance of each feature.

For more information on SHAP values and how they are applied in ML, refer to the work by Lundberg [90], which provide the theory and implementation details needed to apply SHAP values to real-world data sets.

While classical ML models provide structured, rule-based decision making, they lack the ability to handle uncertainty and conflicting information using default logic, which is a crucial aspect of real-world triage. Therefore, a more flexible reasoning system may be required: the BOID. As its reasoning process is fundamentally different from classical ML models, we want to warn the reader that the following section marks a significant shift in concept.

2.5 The BOID Framework

The BOID framework is a decision making model that uses structured prioritization to handle incomplete or conflicting mental attitudes. To do this, the BOID uses default logic. This means that decisions are made based on reasonable assumptions unless contradicted by new evidence [15], which is also how clinicians reason when making triage decisions. In triage, each component plays a vital role: *beliefs* ensure decisions are grounded in accurate, up-to-date information, *obligations* represent the established protocols and guidelines that healthcare providers must follow, and *desires* capture the overarching goals and priorities of healthcare providers, namely, those of patient outcomes. *Intentions* play a distinct role, as they may reflect prior decisions or a default course of action formed through a previous balance of desires, beliefs, and obligations.

The next section explores the principles of default reasoning, followed by a discussion on its relevance in triage, illustrated through an example of BOID reasoning in an ED triaging scenario.

2.5.1 Principles of Default Reasoning

Although developing a full BOID-LLM is beyond the scope of this thesis, we aim to lay the groundwork by exploring whether GPT-4o can infer default rules that align with obligation-based default reasoning patterns found in clinical triage.

Default logics [8] are a form of nonmonotonic reasoning systems [7] that were originally introduced by Reiter [126] in 1980. In contrast to classical logic, where conclusions necessarily follow from premises, default logic allows drawing a preliminary conclusion if it is not identically contradicted by any evidence ("*by default, something is true*"). This logic is fundamental in decision making under uncertainty, especially in contexts such as clinical triage, where complete information is often unavailable and where agents often have to act upon reasonable suppositions or defaults until they are proven wrong.

The basic intuition of default logic is that agents may use default rules to fill in gaps in their knowledge when making decisions. A typical default rule has three parts:

1. **Prerequisite**: The condition for applying the rule.
2. **Justification**: The assumption that must hold unless proven false.
3. **Consequence**: The conclusion drawn if the justification remains unchallenged.

The general form of a default rule is normally written as:

$$\frac{Prerequisite : Justification_1, \ldots, Justification_n}{Consequense} \tag{2.8}$$

where $Prerequisite, Justification_1, ..., Justification_n$ are closed predicate logic formulae, and $n > 0$. Here, the antecedent is the condition that needs to be satisfied for the rule to be applied, the justifications are presumed true unless it is proved otherwise, and the consequences are the conclusions one would draw if no conflict in evidence arises.

To illustrate these concepts, consider the classical example of the Tweety Triangle. Here, we have initial information that *"Tweety is a bird and by default, birds fly"* as well as conflicting information that *"Tweety is a penguin and by default, penguins do not fly"*. In this example, we have the following default rule:

$$\frac{\text{Bird}(x) : \neg\text{Penguin}(x)}{\text{CanFly}(x)} \tag{2.9}$$

where:

1. **Prerequisite**: Bird(x) (If x is a bird ...)
2. **Justification**: $\neg$Penguin(x) (and we can assume x is not a penguin ...)
3. **Consequence**: CanFly(x) (then we can conclude x can fly.)

2.5.2 Relevance and Example in Triage Systems

In triage classification, clinicians have to make decisions based on validated protocols until new information contradicts them. This aligns with default logic, which allows medical professionals to assume certain symptoms indicate specific levels

of urgency unless proven otherwise. Therefore, the BOID framework is highly relevant to emergency triage systems, where immediate and high-stakes decisions must be made despite uncertainty and incomplete information.

Triage decision making often involves conflicts between the different mental attitudes. These mental attitudes can be described using the four components of the BOID framework. To demonstrate how beliefs, obligations, intentions, and desires interact in a triage scenario, consider the following example where a clinician must determine the appropriate KTAS level for a patient presenting with chest pain. Note, however, that this example is a simplification of real-world triage processes, as these often involve a greater number of conflicting rules, both within individual components and across different components of the BOID framework.

*A male patient arrives at the ED presenting with chest pain and stable vital signs. The triage nurse must determine the appropriate KTAS classification based on available information. However, conflicting mental attitudes influence the decision making process. From a **belief**-driven perspective, the nurse is aware that resource constraints in the ED are currently high, and over-triaging stable patients could compromise care for more urgent cases. Therefore, the patient should be classified as KTAS 3. According to established **obligations**, however, triage guidelines dictate that any patient presenting with chest pain should be classified as KTAS 2, ensuring rapid evaluation to rule out potentially life-threatening conditions. **Intentions**, informed by prior triage experience, indicate that many chest pain cases are ultimately found to be non-cardiac in nature. This learned experience leads to a tendency to assign KTAS 3 instead of KTAS 2, particularly when patients present with stable vital signs. Finally, by coincidence, the patient is a family member of the nurse that needs to make a triage classification. Therefore, this nurse has the **desire** to ensure the best possible care for this patient, and would therefore lean toward KTAS 2 instead of KTAS 3.*

This decision conflict can be structured as follows:

$$\texttt{B:} \quad \frac{\text{Chest Pain}(x) : \neg\text{Critical Vital Signs}(x)}{\text{Classified as KTAS Level 3}(x)} \tag{2.10}$$

$$\texttt{O:} \quad \frac{\text{Chest Pain}(x) : \neg\text{Critical Vital Signs(x)}}{\text{Classified as KTAS Level 2}(x)} \tag{2.11}$$

$$\texttt{I:} \quad \frac{\text{Chest Pain}(x) : \neg\text{Critical Vital Signs}(x)}{\text{Classified as KTAS Level 3}(x)} \tag{2.12}$$

$$\texttt{D:} \quad \frac{\text{Chest Pain}(x) : \neg\text{Critical Vital Signs}(x)}{\text{Classified as KTAS Level 2}(x)} \tag{2.13}$$

To resolve this conflict, a BOID priority ordering is established. For example: $\texttt{O} > \texttt{B} > \texttt{I} > \texttt{D}$. In this example, obligations take priority. Therefore, the final KTAS classification is KTAS 2.

However, our KTAS dataset only contains the final triage classifications from registered nurses and experts, without access to their internal decision making processes. Although these classifications may have been influenced by reasoning patterns involving conflicts between beliefs, obligations, intentions, and desires, such cognitive processes are not explicitly recorded. As a result, it is not possible to reconstruct how each BOID component contributed to their final decisions. To address this limitation, we adopt a simplifying assumption: we treat the KTAS classifications as primarily obligation-driven decisions. In other words, we assume that the triage classifications are guided solely by standardized triage protocols and clinical guidelines. Based on this assumption, the present study focuses exclusively on evaluating the O-component of the BOID framework explore whether GPT-4o can simulate obligation-based reasoning.

3 Methods

This section details the methodologies employed in this research. It starts with a description of the original dataset from Kaggle [182]. Then, the data preprocessing pipeline is described, showing the steps we have taken to form the final dataset used for this research. Here, we also provide an overview of all elements in the data as well as an example case. After this, we show how we addressed the inherent class imbalance in the KTAS dataset.

The section proceeds with an overview of the feature descriptions and their distributions. Next, the implementation of the classical ML models (DT, RF, and XGBoost) is described, including hyperparameter tuning, cross-validation strategies, and feature importance analysis using SHAP values.

Following, we will show the implementation of GPT-4o for KTAS-level prediction. This section details the prompt to extract triage decision rules from GPT-4o as well as the prompt engineering techniques employed to investigate the model's reasoning.

The section ends by defining the evaluation metrics used to compare model performance.

3.1 Original Data Description

We used an open-source dataset from the Kaggle repository, which was contributed in South Korea [182]. It consisted of 1540 systematically selected medical records of adult patients admitted to two EDs, one local and one regional, between October 2016 and September 2017. After excluding the records with a triage duration of more than 30 minutes, were canceled at reception before triage, and had incomplete data to evaluate true triage, in total 1267 eligible medical records remained.

Each record contains 24 variables, including patient demographics and vital signs, which were documented during the initial nursing assessment. However, not all of these variables were retained in the final dataset. A full overview of the final data structure, including the selected variables, as well as an example case, will be provided in the next section, in which we detail the preprocessing steps.

Each patient was assigned two KTAS urgency scores: the first established by registered emergency nurses and the second, which is considered the gold standard, by three qualified triage experts: a certified emergency nurse, a KTAS provider and instructor, and an experienced emergency nurse recognized for their competence. This unique structure, which motivated our choice of this dataset, allows us to explore how different clinical perspectives inform triaging decisions. The process of assigning their KTAS labels is illustrated in Figure 2.1 in Section 2.1.

3.2 Data Preprocessing

Data preprocessing is a fundamental step in the data analysis pipeline, as it ensures that the data is clean, consistent, and ready to be used in ML models. For this study, the preprocessing pipeline was designed to handle the challenges posed by the KTAS dataset. Below is a detailed explanation of the preprocessing steps carried out and the rationale behind them.

The preprocessing and analysis were conducted in a Python-based environment. Details about the specific computational setup are provided in Appendix A.

Step 1: Mapping Free-Text Complaints to Standardized Categories Patient complaints recorded as free text were mapped to predefined categories from Aronsky et al. [11], using a regex-based dictionary. This approach allowed the grouping of semantically similar complaints under standardized labels. For example, terms like "abd pain," "abdominal pain," and "RLQ pain" were mapped to the "Abdominal Pain" category. This transformation was critical to reduce dimensionality and improving interpretability through structured labels. The mapping function was applied to the `Chief complain` column, and the original column was dropped after the transformation to avoid redundancy. The newly created column reduced the original number of complaint types from 417 to 43.

Step 2: Handling Missing Values Columns `SBP` (Systolic Blood Pressure), `DBP` (Diastolic Blood Pressure), `HR` (Heart Rate), `RR` (Respiratory Rate), `BT` (Blood Temperature), and `Saturation` contained missing data. These were replaced with `NaN` to ensure compatibility with Python's data manipulation tools. Table 3.1 presents the number of missing values per column. We want to highlight that more than half of the patient entries (54.91%) lacked a recorded `Saturation` value. Despite this, we retained this feature due to its possible critical role in assessing patient acuity, as oxygen saturation levels are, for example, essential indicators of mortality during hospitalization [56] and the disease course of patients [17].

To address these missing values, we employed mode imputation, grouping by `expert KTAS` categories to ensure that the imputed values align with typical vital signs levels within each triage level. This approach follows the methodology presented in the Kaggle notebook "Triage Application with Machine Learning Models" by Yıldız [174], which utilized the same dataset to predict the reason for mistriage by registered nurses. However, while their method grouped by both `expert KTAS` and `mistriage`, we grouped only by `expert KTAS`, as our study does not focus on mistriage reasons.

Column	**Number of Missing Values (NaN)**	Percentage (%)
SBP	25	1.98%
DBP	29	2.29%
HR	20	1.58%
RR	22	1.74%
BT	18	1.42%
Saturation	694	54.91%

Table 3.1: Number of Missing Values

Step 3: Removing Irrelevant Features Columns `Error_group`, `Length of stay_min`, `KTAS duration_min`, `mistriage`, `Disposition`, and `Diagnosis in ED` were considered irrelevant for the study's objectives and dropped. The `Error_group` column represents the overarching classifications of the underlying causes for mismatched KTAS scores (e.g. inadequate application of the pain scale to the KTAS algorithm), as determined by the triage experts. As we are not analysing the reasons behind mistriage, but focusing on classifying patients based on demographics and clinical variables to evaluate decision making frameworks, we removed this column. Furthermore, the `Length of stay_min` column was removed because it represents a post-triage outcome rather than an input variable relevant to the decision making process at the time of triage. Including this column could introduce data leakage, as it reflects information that would not be available to registered nurses or triage experts when making the initial KTAS classification. For the same reason, we also removed the `KTAS duration_min` column.

The `mistriage` column was removed as well, because these values reflect an evaluation of the triage process rather than an independent variable available at the time of triage. Specifically, this column indicates whether the triage decision made by the registered nurses deviated from the experts' assessment, effectively serving as an outcome variable rather than an input. Including it could inadvertently introduce circular reasoning or bias into the analysis, as it would provide a retrospective judgment of the triage classification. This is the same reason why we exclude the `Disposition` and `Diagnosis in ED` columns.

Step 4: Resolving Data Inconsistencies To address formatting issues, commas in numerical values were replaced with periods to align with Python's numeric parsing. Column names were cleaned by removing double quotes to ensure compatibility with Python's syntax. These transformations ensured that the dataset was free from formatting anomalies that could hinder downstream operations, such as numerical computations or column indexing.

Step 5: Validating and Correcting Pain Columns We found that instances with `Pain == 0` always occured together with `NRS pain == #BOƤ!`, with NRS (numerical rating scale) pain ranging from 0 to 10. Therefore, we replaced these values with `0`. Instances where `Pain == 1` but `NRS pain == 0` were removed, as these are invalid combinations. This resulted in a total of 1264 data entries, as was the case for three patients. This validation step ensured data integrity. However, we kept the `Pain` column in our dataset, as its binary nature provides a simple high-level indicator that can serve as a feature for rules or thresholds within the model.

Step 6: Converting Numeric-Like Columns Columns `SBP`, `DBP`, `HR`, `RR`, `BT`, `Saturation`, and `NRS pain` were converted to numeric types using `pd.to_numeric`, ensuring proper handling of non-numeric placeholders. This step avoided potential errors during statistical computations or model training.

Step 7: Transforming Categorical Variables Categorical variables `Arrival mode` and `Mental` were mapped to descriptive labels using dictionaries. The `Mental` column originally consisted of integer values, representing distinct categorical states (1: Alert, 2: Verbal Response, 3: Pain Response, 4: Unresponsive). Allowing DT-based models to split on numeric thresholds like >1.5 would not

have logical meaning, as integers do not have an ordinal relationship or numerical continuity. By mapping these values to descriptive strings, we clarified their categorical nature, enabling the models to handle them correctly during training through techniques such as one-hot encoding. Similarly, the `Arrival mode` column was updated to replace integer codes with descriptive categories (1: Walking, 2: Public Ambulance, 3: Private Vehicle, 4: Private Ambulance, 5: Other, 6: Other, 7: Other). Integers 5, 6, and 7 all corresponded with "other" modes of arrival, so we grouped these together to the value 'Other'. This transformation improved interpretability by replacing numeric codes with human-readable labels and ensured compatibility with downstream feature engineering.

Table 3.2 shows an overview of the description of the dataset before scaling the numerical features. To provide an example, consider a 54-year-old alert female patient who arrived at the regional ED via private ambulance. Her chief complaint was fever. She reported no injury or pain, with a recorded pain level of 3 of out 10. Her vital signs were as follows: systolic blood pressure 140.0 mmHg, diastolic blood pressure 90.0 mmHg, heart rate 94.0 bpm, respiratory rate 20.0 breaths per minute, body temperature 38.1°C, and oxygen saturation 98.0%. In the same hour of her arrival, six other patients had also arrived. Using these variables, the most appropriate KTAS level according to registered nurses was 3, while according to experts, this patient better fits within the urgency level 4.

Step 8: Scaling Numerical Features We employed robust scaling [134] with 5th and 95th percentiles to preprocess numeric variables. It moves data towards the median and scales based on the interquartile range (IQR). This makes it less sensitive to outliers, while maintaining the natural relative relationships of data points. By utilizing the 5th and 95th percentiles, we encompassed more of the data so that the scaling was more appropriately calibrated to the variation of the clinical measurements. We were subsequently able to maintain the integrity of the key data points while not allowing the extreme outliers to dominate, and thus we obtained a more stable and uniform model performance.

Step 9: Encoding Categorical Variables The transformed categorical columns (`Arrival mode`, `Mental`, and `complaint_category`) were one-hot encoded. This encoding method represented each category as a binary variable, ensuring compatibility with ML algorithms that require numeric input.

In summary, the preprocessing pipeline was designed to address the challenges of the KTAS dataset. The structured and comprehensive approach allowed for the creation of a clean, interpretable, and ready for analysis dataset.

3.3 Data Balancing

The KTAS dataset presents a notable class imbalance, particularly in the most critical (KTAS 1) and least critical (KTAS 5) classes (see Tables 3.16 and 3.17). This imbalance poses a risk of model bias towards the majority classes (KTAS 3 and 4), potentially reducing the model's ability to accurately predict underrepresented but clinically critical cases.

Feature Name	Data Type	Possible Values (Mean ± SD)	Percentage (%)
`Sex`	Numerical	1: female	47.79%
		2: male	52.22%
`Age` (Years)	Numerical	54.36 ± 19.70	-
`Group`	Numerical	1: local ED	54.19%
		2: regional ED	45.81%
`Patients number per hour`	Numerical	7.51 ± 3.16	-
`Arrival mode`	Categorical	1: walking	6.25%
		2: public ambulance	20.89%
		3: private vehicle	59.49%
		4: private ambulance	12.26%
		5: other	1.11%
`Injury`	Numerical	1: no	80.78%
		2: yes	19.23%
`Pain`	Numerical	0: no	43.75%
		1: yes	56.25%
`NRS pain` (Numerical Rating Scale pain)	Numerical	0 (no pain)	43.75%
		1	0.16%
		2	3.01%
		3	21.99%
		4	11.16%
		5	10.76%
		6	5.54%
		7	2.61%
		8	0.71%
		9	0.08%
		10 (max pain)	0.24%
`Complaint category`	Categorical	See Table 3.3	-
`Mental status`	Categorical	1: alert	93.67%
		2: verbal response	3.09%
		3: pain response	2.22%
		4: unresponsive	1.03%
`SBP` (Systolic Blood Pressure) in mmHg	Float	132.82 ± 28.32	-
`DBP` (Diastolic Blood Pressure) in mmHg	Float	79.36 ± 15.93	-
`HR` (Heart Rate) in bpm	Float	84.05 ± 16.49	-
`RR` (Respiratory Rate) in breaths/min	Float	19.51 ± 1.98	-
`BT` (Blood Temperature) in °C	Float	36.57 ± 0.54	-
`Saturation` in %	Float	97.51 ± 1.991	-
`registered nurse KTAS`	Numerical	1 (most urgent)	1.42%
		2	16.77%
		3	35.29%
		4	39.64%
		5 (least urgent)	6.88%
`expert KTAS`	Numerical	1 (most urgent)	1.98%
		2	17.41%
		3	38.37%
		4	36.31%
		5 (least urgent)	5.94%

Table 3.2: Description of the Dataset with $n = 1264$

To address this issue, we employed Adaptive Synthetic Sampling (ADASYN) [53], an oversampling technique that generates synthetic samples based on the distribution density of minority class instances. We implemented ADASYN with the `imblearn.over_sampling.ADASYN` function from the imbalanced-learn package [77]. Unlike SMOTE (Synthetic Minority Over-sampling Technique) [19], which generates synthetic samples uniformly, ADASYN prioritizes harder-to-classify instances, adding more synthetic samples where the data is most sparse. This targeted approach helps refine decision boundaries, ultimately enhancing the model's performance in recognizing rare but clinically significant cases.

In our study, ADASYN was implemented in the training set, replicating KTAS 1 and KTAS 5 to 150 instances each. This threshold was selected empirically

through trying various resampling ratios and evaluation their influence on model performance. By providing the model with an improved distribution of the extreme triage levels, ADASYN reduces classification bias, causing the model to classify life-threatening emergencies (KTAS 1) as well as non-emergency cases (KTAS 5) more effectively.

3.4 Feature Description and Distribution

The dataset has a comprehensive set of variables. These variables play a crucial role in understanding patient distribution and the association with urgency levels as captured by the KTAS system.

Demographics The age distribution of the patients (Figure 3.1) ranges from young adults to elderly patients. The average age is 54.36 (with SD 19.70), as can be seen in Table 3.2. The sex distribution (Figure 3.2) is fairly balanced, with a slight overrepresentation of men (660) compared to women (604).

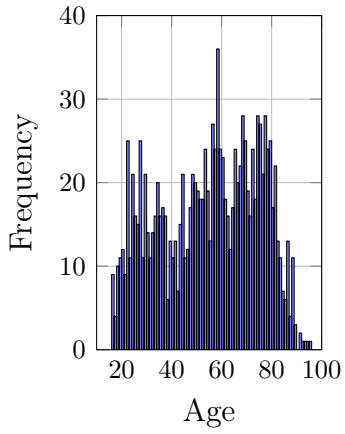


Figure 3.1: Distr. of Age

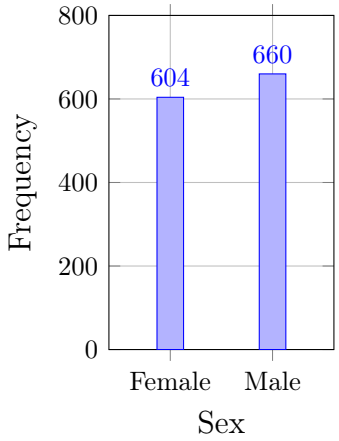


Figure 3.2: Distr. of Sex

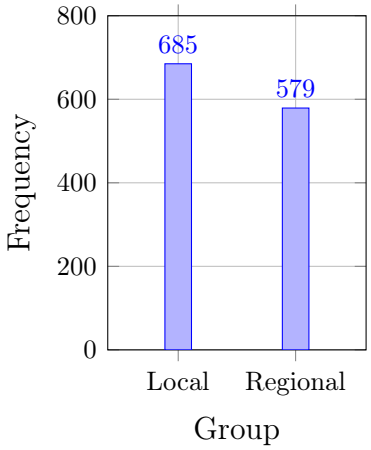


Figure 3.3: Distr. of Group

Vital Signs The distribution for the variables `SBP` (Systolic Blood Pressure), `DBP` (Diastolic Blood Pressure), `HR` (Heart Rate), `RR` (Respiratory Rate), `BT` (Blood Temperature), and `Saturation` are displayed before scaling the values according to step 8 from the preprocessing pipeline to be able to give more understandable insights into the distribution of these variabeles. Figures 3.4, 3.5, and 3.6 show the distributions of `SBP`, `DBP`, and `HR`, and Figures 3.7, 3.8, and 3.9 show the distributions of `RR`, `BT`, and `Saturation`. As shown in the plots, the vital sign distributions vary in shape and spread, but they all exhibit relatively normal distributions centered around typical physiological values [23].

Group The group distribution illustrates the proportion of patients categorized into Local ED and Regional ED groups. As shown in Figure 3.3, there is a slightly higher representation of Local ED cases.

Patients number per hour Figure 3.10 represents the number of patients who arrive at the same hour as the corresponding patient. The distribution reveals that most patients arrive during hours when five to nine other patients also present.

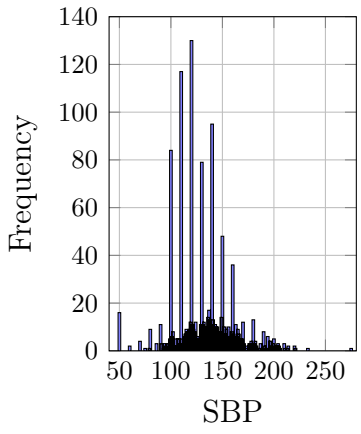


Figure 3.4: Distr. of SBP

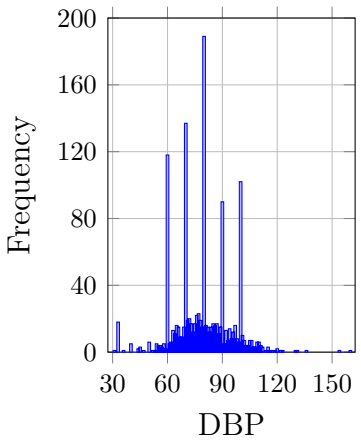


Figure 3.5: Distr. of DBP

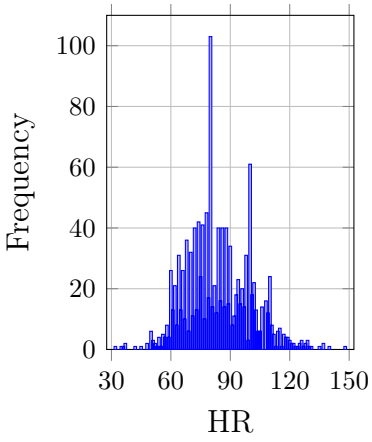


Figure 3.6: Distr. of HR

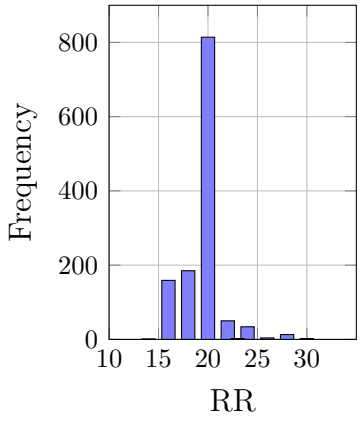


Figure 3.7: Distr. of RR

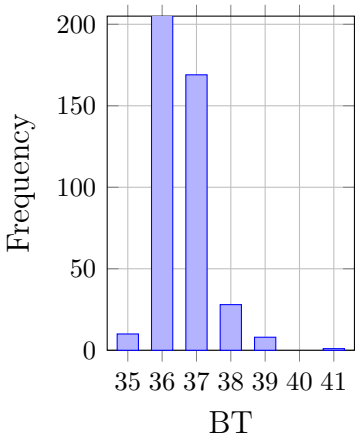


Figure 3.8: Distr. of BT

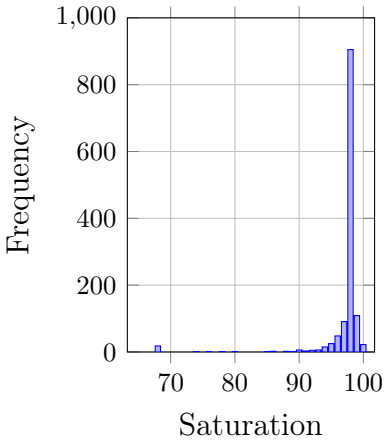


Figure 3.9: Distr. of Saturation

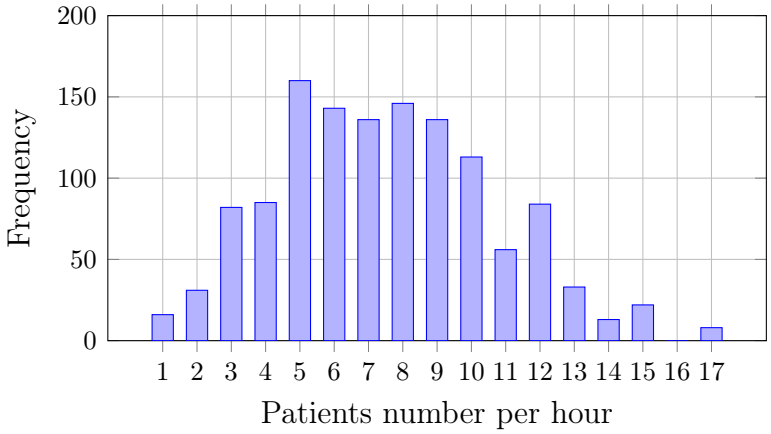


Figure 3.10: Distribution of Patients Number per Hour

Arrival Mode Figure 3.11 provides information on the arrival mode of the patient. Most of the patients (752) arrive by private vehicles, followed by public

ambulances (264) and private ambulances (155). Patients who walk (79) and those who arrive by other means (14) represent the minority.

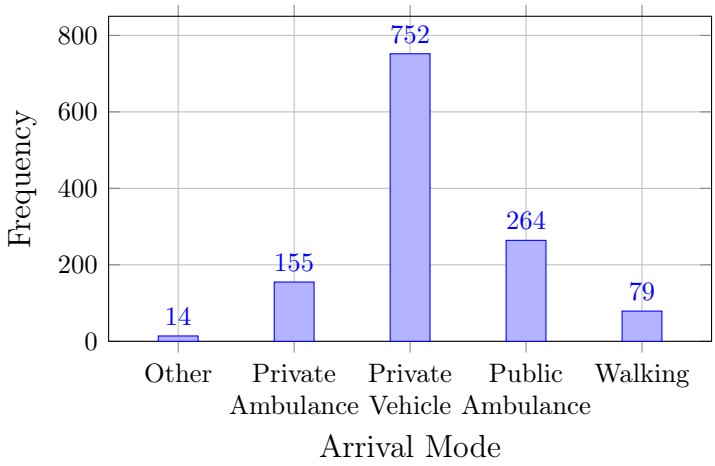


Figure 3.11: Distribution of Arrival Modes

Injury and Pain Figure 3.12 highlights that injuries are relatively rare, with only 243 patients reporting injuries compared to 1,021 who did not. As illustrated in Figure 3.13, most of the patients (711) reported experiencing pain, while 552 patients presented without pain.

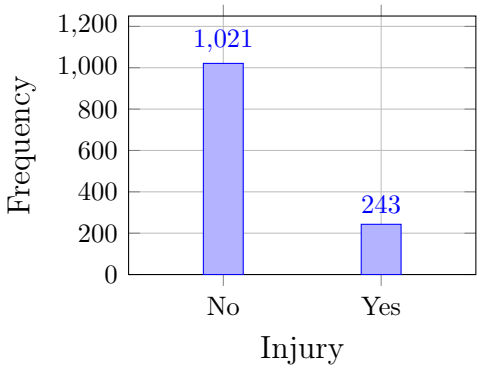


Figure 3.12: Distribution of Injury

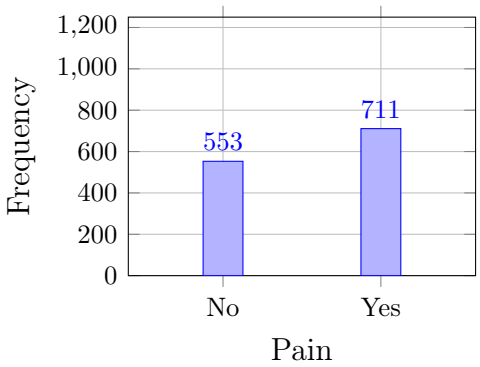


Figure 3.13: Distribution of Pain

NRS Pain Pain severity is measured using the NRS (Numerical Rating Scale), which ranges from pain level 0 to 10, with Figure 3.14 showing a wide distribution across the scale. The majority of patients (553) report no pain (NRS=0), while 278 report mild pain (NRS=3). Severe pain is less frequent. Summing all pain scores with values greater than or equal to 1 reveals that 711 patients reported experiencing pain, a finding consistent with the distribution shown in Figure 3.13.

Complaint Category Table 3.3 shows that the newly created `complaint category` column holds the standardized category name for each patient complaint, with "Abdominal Pain" and "Chest Pain" being the most common complaints.

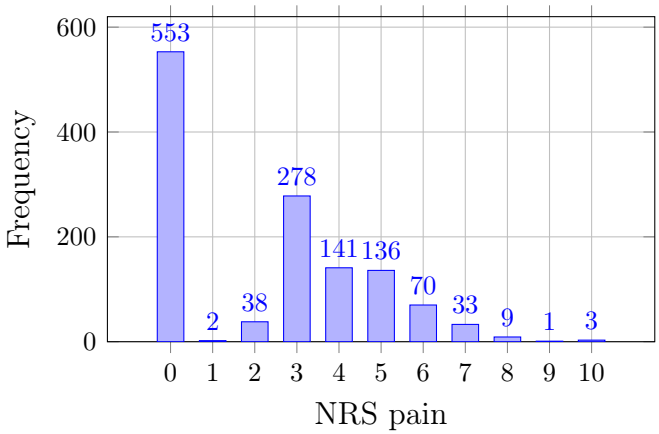


Figure 3.14: Distribution of NRS pain

Complaint Category	Freq.	Complaint Category	Freq.
Abdominal Pain	185	Back Pain	18
Chest Pain	98	Ear/Nose/Throat Problems	16
Traumatic Injuries Specific (FT)	73	Convulsions; Seizures	15
Dizzy	69	Procedure	14
Respiratory Problems	66	Fainting/Syncope	12
Hemorrhage	62	Dental; Toothache	9
Weakness	61	Foreign Body	8
Fever	58	Unconsciousness	8
Abdominal Problems	49	Substrance Abuse	6
Psychiatric/Social Problems	48	Body Aches	6
Headache	48	Follow-Up	6
Eye Problem	40	Burns	5
Peripheral Vascular/Leg Pain	38	Neck Pain	5
Flank Pain	33	Diabetic Problems	5
Neurological Complaint	33	(Specific) diagnosis (FT)	4
Laceration	27	Medication refill	4
Skin Complaint/Trauma	23	Assault/Rape	4
Unknown Problem	21	Cardiac Arrest	4
Genito-Urinary Problem	21	Allergies/Hives/Med Reaction/Sting	3
Orthopedic Injury	20	Obstetrical Problem	1
Fluid/Nutrition Alternation	19	Gynecological Problem	1
Cardio-Vascular Complaint	18		

Table 3.3: Categories and Their Frequencies

Mental Status Figure 3.15 highlights the predominance of patients presenting as alert (1,184), with smaller subsets exhibiting responses to pain (28), being unresponsive (13), or showing verbal responses (39).

KTAS Registered Nurse vs. Expert The KTAS distributions for both registered nurses and experts evaluations (Figures 3.16 and 3.17) demonstrate a notable similarity, with the majority of patients classified as KTAS levels 3 and 4. Level 3 is the most common, encompassing 501 cases by registered nurses and 485 by experts. Levels 1 and 5 are less frequent, with 18 and 87 cases respectively for nurses, compared to 25 and 75 for experts.

The confusion matrix in Table 3.18 illustrates the agreement between registered nurses and experts KTAS assignments, showing a high concentration of cases along the diagonal. We see general alignment between the two classifications. However, discrepancies are evident, suggesting differences in interpretation or assessment

criteria. This is particularly relevant to our research question, as we aim to investigate whether ML models align more with registered nurses-assigned KTAS levels or expert-assigned KTAS levels.

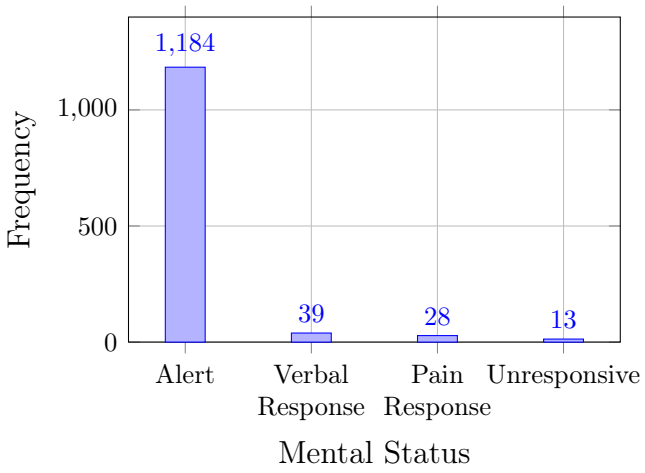


Figure 3.15: Distribution of Mental Status

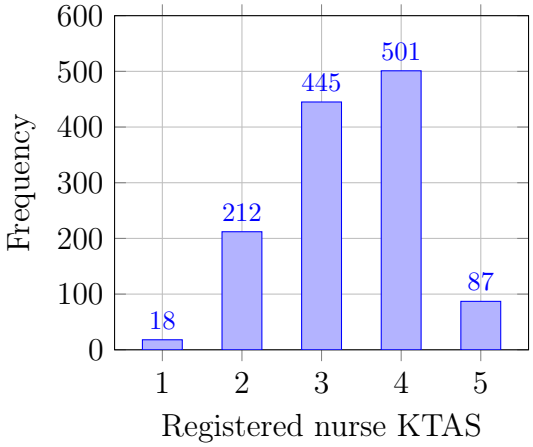


Figure 3.16: Distr. of RN KTAS

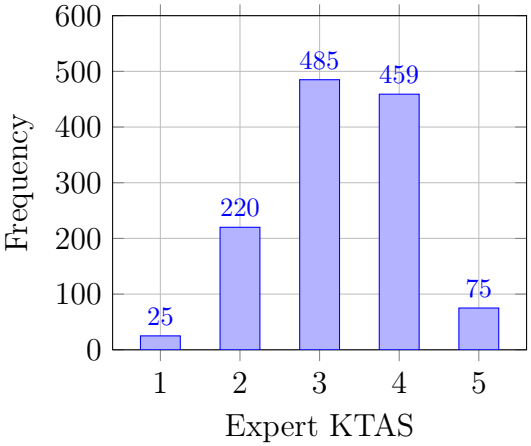


Figure 3.17: Distr. of Expert KTAS

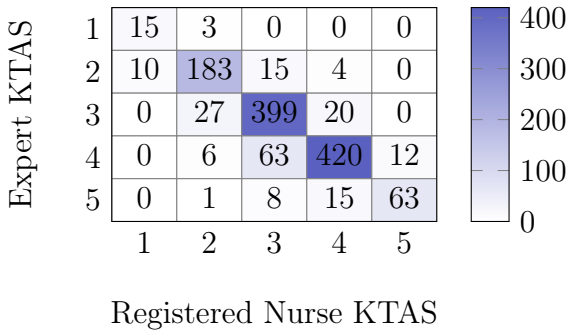


Figure 3.18: Confusion Matrix for Registered nurse KTAS vs Expert KTAS

To further understand the dataset, exploratory data analysis (EDA) was conducted. Since EDA was not directly used for feature selection or model modifications, detailed results are provided in Appendix B.

3.5 Implementation of Classical ML Models

To ensure a reliable evaluation of model performance, we divided the dataset into a training set and a test set using a 75/25 split. This resulted in 948 patients used for training and 316 patients used for final testing. The test set was held out throughout the training process and only used for evaluating the generalization performance of the final models. You can reach out to Figure 1.2 for the total modeling design for KTAS prediction task for classical ML models. As also illustrated, each model was trained and tested on either the expert-assigned KTAS labels or the registered nurse-assigned KTAS labels.

3.5.1 Hyperparameter Tuning and Cross-Validation

To optimize the DT, RF, and XGBoost models, we employed GridSearchCV [74], that systematically searches a predefined hyperparameter spaces to identify the best performing configurations. After trial and error, we established the hyperparameter spaces. They are displayed in Table 3.4. Key parameters such as tree depth, split criteria, number of estimators, and learning rate were fine-tuned to enhance predictive performance.

While the default scoring metric in GridSearchCV is accuracy, accuracy can be misleading in imbalanced datasets [41]. In such cases, a model may achieve high accuracy by simply predicting the majority class. With our dataset, this would result in models to predict KTAS levels 3 and 4 more often than the most or least urgent KTAS levels 1 and 5 (see Figures 3.16 and 3.17), while these minority classes are of great importance. Therefore, we selected the weighted F1 score as the scoring metric. This metric provides a balanced measure of precision and recall across all KTAS levels and ensures that model performance reflects the imbalanced distribution of urgency classes [60].

To further address class imbalance, we used StratifiedKFold cross-validation [75], which maintains the original class distribution in each fold. This prevents the model from being trained on unrepresentative data subsets. We selected 5-folds to balance between computational efficiency and reliable evaluation.

3.5.2 Interpretability through SHAP Analysis

Since we are investigating the performance of DTs, RFs, and XGBoost, we implement TreeSHAP by calling the `shap.TreeExplainer` function from the SHAP library [91]. This is an optimized variant of SHAP specifically designed for tree-based models [92]. TreeSHAP computes SHAP values in polynomial time in an efficient manner, making it considerably faster and scalable than standard SHAP implementations. By applying TreeSHAP to our models, we aim to understand

Model	Hyperparameters Selected for Training
Decision Tree	Criterion: Gini, Entropy Maximum Depth of Tree: [5, 10, 15, 20, 25, 30, 35, 40]. Minimal Samples Leaf of Tree: [1, 2, 4, 6, 8, 10]. Minimal Samples Split of Tree: [2, 5, 10, 15, 20, 25, 30].
Random Forests	Criterion: Gini, Entropy. Amount of Estimators: [50, 250, 500]. Maximum Depth of Tree: [1, 3, 5, 7, 9, 11, 13, 15]. Minimal Samples Leaf of Tree: [1, 2, 4, 6, 8, 10, 12]. Minimal Samples Split of Tree: [2, 4, 6, 8, 10, 12].
XGBoost	Amount of Estimators: [50, 250, 500, 1000]. Maximum Depth of Tree: [1, 5, 10, 15, 20, 25, 30]. Minimum Weight of Child: [1, 3, 5, 7]. Learning Rate: [0.01, 0.05, 0.1, 1, 10]. Gamma: [0, 0.1].

Table 3.4: Set of Hyperparameters for each Classical ML Model

how clinical features influence the model's assignment of KTAS levels, either increasing or decreasing their contribution to a particular prediction. To visualize the results, we generated SHAP summary plots.

3.6 Implementation of GPT-4o

GPT-4o was accessed via the OpenAI API, using the `chat.completions.create` function [113]. The temperature parameter controls the amount of randomness in the output of the model, of which the value may range between 0 and 2. Lower values make responses more deterministic, and higher values introduce more variability [28]. Given the need for reliable and reproducible predictions in triage scenarios, a temperature of 0 was chosen, minimizing response variability and ensuring consistent responses of GPT-4o.

The design for the KTAS prediction task for GPT-4o using the various prompting techniques can be found in Figure 1.3. As you can see, GPT-4o received no direct access to the KTAS dataset during inference. Instead, predictions were generated based solely on the patient information provided in the prompt, combined with the model's pretrained knowledge. The resulting KTAS level predictions were evaluated against the expert KTAS labels, which serves as the gold standard.

3.6.1 Triage Rule Extraction

To evaluate the ability of GPT-4o to generate clinically meaningful triage rules, we used trial and error. The final prompt can be found in Appendix C.1. We prompted the model to produce a comprehensive, hierarchical, and logically consistent set of KTAS rules. The steps in this prompts emphasized mutual exclusivity, collective exhaustiveness, prioritization of higher-acuity cases, and inclusion of multiple clinical conditions per rule to reflect real-world complexity. Besides, the model was also instructed to allow exceptions to capture the default-based reasoning performed by clinicians when using decisions. Furthermore, GPT-4o was prompted to assume that it was an AI expert in emergency medicine that should

generate structured `IF-THEN` rules, explicitly defining the classification criteria for each KTAS level. Also, we instructed the model to output only the list of rules, maintaining a clear output.

For generating the rules, we explained how our KTAS dataset is built, providing explanations about all variables and their possible values. However, we did not provide any KTAS-specific examples or access to external tools, as we wanted to assess to what extent the inferred triage rules of GPT-4o align with the gold standard expert KTAS labels, solely relying on its internal knowledge.

3.6.2 Prompt Engineering Techniques

For each prompting technique, we followed a general workflow (see Algorithm 1), ensuring consistency between experiments. To prevent context length from becoming too large, which could lead to information loss or reduce prediction accuracy [79, 81, 136], we chose to divide patient data into batches of 10. By generating predictions for groups of 10 patients instead of tackling each one separately, we were able to manage costs effectively.

In our experiments, we used of both system and user prompts to guide GPT-4o's responses. The system prompt acts as the main instruction, outlining the model's role, context, and the output standards [133]. Meanwhile, the user prompt includes task-specific input, incorporating structured patient data [133]. For almost every prompt, the user prompt was a list formatted in JSON, but for the fluent style prompt, we presented the patient information in a more natural, conversational text to investigate how this would influence GPT-4o's predictions. Despite this variation, everything else remained unchanged. We prompted the model to return a JSON list containing the KTAS predictions for each patient in the batch. The full code for the general workflow can be seen in Appendix C.2.

Algorithm 1 GPT-4o General Workflow

```
Input: Patient dataset test set
Define system_prompt as predefined system message (see Appendix C.3)
for each batch of 10 patients in patient dataset do
    Extract batch from dataset
    Format user_prompt as JSON string with patient data
      # user_prompt is different for fluent style prompting, see Appendix C.7
    Call GPT-4o API with:
      - system_prompt
      - user_prompt
    Extract model response
end for
Output: GPT-4o predictions for each batch
```

The diversity of prompt formulations allowed us to investigate how prompts affect GPT-4o's predictive performance in our triage classification task.

3.6.2.1 Standard Prompting

We employed the *zero-shot prompting* technique as baseline to evaluate GPT-4o's KTAS classification performance [84]. It leverages the model's pre-trained knowledge without providing examples but asked the model to deduce classifications based only on the provided patient data. The *few-shot prompt* augmented this by including two example cases along with corresponding KTAS classifications, demonstrating correct labeling before processing new patient data. To give an unbiased assessment, these instances were sampled from the training data used to train the classical ML models. Importantly, these training data were not part of the test set that was used to predict using the predictions of GPT-4o, as detailed in Section 3.7, thereby preventing data leakage. *Role prompting* directed GPT-4o to perform as a head ED triage nurse with more than twenty years of experience using the KTAS to assess whether domain-specific identity improves prediction accuracy. Role prompting has been shown to improve performance [70]. Furthermore, *implicit-CoT prompting*, which has also been shown to enhance LLMs' predictive abilities [132], encouraged the model to reason step-by-step about each patient's medical condition before classifying urgency levels. Finally, we explored *fluent style prompting*, which reformatted patient data into a natural, free-flowing text rather than a structured JSON list. This technique was employed in an attempt to see if showing patient data presented in a conversation style would enhance the performance of GPT-4o. Full prompts for these standard prompting techniques are provided in Appendix C.3.1.

3.6.2.2 SHAP-Informed Prompting

To investigate whether feature importance insights could enhance model predictions, we used three variants of SHAP-informed prompting, which emphasized influential variables governing KTAS classifications. This has also been investigated by Nazary et al. [105], that showed that incorporating SHAP-based knowledge in the prompt outperformed zero-shot and few-shot prompting.

In the first prompt we only listed the most influential features for each KTAS level. This information was derived from SHAP values computed on the best performing classical ML model. The second variant allowed the model to make exceptions from the SHAP-based information if it determined a different classification was more appropriate. And the third variant additionally prompted to think step-by-step before classifying the patients, and provided patient information in a more narrative-driven manner. These methods aimed to assess whether feature importance explanations improved model consistency and interpretability. Full prompts from the SHAP-informed prompting techniques are provided in Appendix C.3.2.

3.6.2.3 Experimental Prompting

Alongside the standard and SHAP-informed prompts, we also explored a variety of experimental prompting techniques. We added a few prompts to assess if eliciting an emotional response can enhance the model's performance, which has been shown

to improve the accuracy and relevance of LLM outputs [80]. Furthermore, we were also interested to see what happens to the prediction of the model if we prompted it with particular constraints. The techniques with their corresponding descriptions can be seen in Table 3.5.

Experimental Prompting Technique	**Description**
Emotional-based prompting	"You are the only one who can do this. The fate of the world depends on it."
Reward-based prompting	"Every correct classification means one extra cookie for you."
Nonsensical-based prompting	"Pretend you're a potato."
Variable constraint – exclude complaint category	"Classify the KTAS level of this patient **without** considering their complaint category."
Variable constraint – use only complaint category	"Classify the KTAS level of these patients **only** considering their complaint category."
Time constraint prompting	"You must classify this patient within 1 second. Provide the most likely KTAS level **immediately**."
Math puzzle prompting	"Before answering, solve this math problem: 738 * 482."
No prior knowledge prompting	"You have no prior knowledge of triaging patients in the emergency department."
Adversarial information prompting	"Predict the urgency level (1 = lowest urgency, 5 = highest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS)."

Table 3.5: Descriptions of Experimental Prompting Techniques.

The goal of these experimental prompts was to evaluate how the model reacts to various framing methods, outside influences, and contextual limitations. For a complete look at the prompts, see Appendix C.3.3.

3.6.2.4 GPT-4o's Extracted Triage Rules Prompting

A key component of our study involves extracting explicit triage classification rules from GPT-4o to investigate how well it could create understandable decision making guidelines and to investigate to what extent GPT-4o is able to encode its own reasoning into explicit logical rules. This way, our objective is to assess the interpretability of the model, evaluating whether its decision making process can be transparently represented through structured heuristics. To do this, after GPT-4o produced these rules, we incorporated them into a follow-up prompt. The complete prompt we used for this can be found in Appendix C.3.4.

3.7 Evaluation Metrics

The task in this study is to classify patients into one of five levels of the KTAS, making this a classification problem. Because of this, we use established evaluation metrics for classification tasks, specifically accuracy and F1 score, as these are widely accepted in classification performance assessment [58]. The first evaluation metric is accuracy. This measures the total correctness of the predictions of the trained ML model on the test data. The F1 score provides us with the recall-precision balance since it measures the harmonic mean of the two. This is

most valuable in imbalanced class distribution cases, providing a more accurate evaluation than accuracy measures alone [4].

However, KTAS is not just any classification task, since it is an ordinal classification problem, where the class labels are ordered in terms of clinical urgency. This ordinal nature, combined with class imbalance in the data, makes it helpful to complement traditional metrics with additional ones, such that we capture a holistic understanding of the performance of our models. Therefore, we also used Cohen's Kappa and Cohen's Weighted Kappa. Cohen's Kappa evaluates the agreement between predicted and actual classifications, adjusting for chance agreement. The weighted version is an extension of this, with allowance for the extent of disagreement, especially relevant for ordinal classification tasks like ours [150]. For example, in the KTAS system, misclassifying a patient with a critical KTAS level 1 as level 2 is less severe than misclassifying them as level 5, and weighted Cohen's Kappa accounts for this ordinal severity by penalizing larger misclassifications more heavily. So, these metrics provide insight into the model's reliability and its ability to make correct predictions. Besides these, we also selected the AUC metric, which measures a model's ability to distinguish between classes across various threshold settings, with an AUC value closer to 1 indicating superior discriminative performance. This metric is crucial to assess how well the model can separate different classes, which is essential in our KTAS classification task [4].

Furthermore, to ensure a fair comparison between our classical ML models and GPT-4o, all models were evaluated using the same test dataset, consisting of 316 patients. This dataset remained unchanged in all tests. Additionally, when using few-shot prompting in GPT-4o, the two example cases were extracted from the training data to ensure that they were not present in the test dataset. These precautions prevented inconsistency and data leakage.

4 Results

This section presents the findings from our experiments, covering both classical ML models and GPT-4o's performance in KTAS classification. We begin with the performance of DT, RF, and XGBoost, assessing their ability to predict KTAS levels assigned by both registered nurses and triage experts. Following this, we display the performance of GPT-4o, showing how different prompt engineering techniques impact its ability to classify KTAS levels and whether the model can extract and apply its own triage rules.

4.1 Results of Classical ML Models

This section presents the performance of our DT, RF, and XGBoost models in predicting KTAS assignments assigned by registered nurses and triage experts and shows how model performance varies between different KTAS levels between these two groups. An interpretation on these results is provided in Section 5.1.1, that discusses the results. Refer to Table 4.1 for an overview of the performance of all models in predicting registered nurse KTAS and expert KTAS. Furthermore, Table 4.2 displays the optimal hyperparameters for predicting registered nurse KTAS, while Table 4.3 details the best hyperparameters for predicting expert KTAS, both across all models. The best value for each evaluation metric is shown in bold.

Task	**Model**	**Accuracy**	**Cohen's Kappa**	**Weighted Kappa**	**AUC**
Registered nurse KTAS	DT	0.6108	0.4403	0.5091	0.7730
	RF	**0.6487**	**0.4680**	**0.6020**	**0.8799**
	XGBoost	0.6361	0.4550	0.5850	0.8444
Expert KTAS	DT	0.6551	0.4846	0.6328	0.7963
	RF	**0.6994**	**0.5448**	0.6538	**0.8941**
	XGBoost	0.6930	0.5410	**0.6884**	0.8819

Table 4.1: Performance Classical ML models for Registered Nurse KTAS and Expert KTAS Predictions

Model	Best Hyperparameters for predicting registered nurse KTAS
Decision Trees	Criterion: Gini. Maximum Depth of Tree: 20. Minimal Samples Leaf of Tree: 1. Minimal Samples Split of Tree: 30.
Random Forests	Criterion: Gini. Amount of Estimators: 250 Maximum Depth of Tree: 20. Minimal Samples Leaf of Tree: 1. Minimal Samples Split of Tree: 5.
XGBoost	Amount of Estimators: 250. Maximum Depth of Tree: 15. Minimum Weight of Child: 1. Learning Rate: 0.1. Gamma: 0.1.

Table 4.2: The best hyperparameters for predicting registered nurse KTAS for each model

Model	Best Hyperparameters for predicting expert KTAS
Decision Trees	Criterion: Gini. Maximum Depth of Tree: 15. Minimal Samples Leaf of Tree: 1. Minimal Samples Split of Tree: 30.
Random Forests	Criterion: Gini. Amount of Estimators: 500. Maximum Depth of Tree: 15. Minimal Samples Leaf of Tree: 1. Minimal Samples Split of Tree: 2.
XGBoost	Amount of Estimators: 50. Maximum Depth of Tree: 1. Minimum Weight of Child: 3. Learning Rate: 1. Gamma: 0.

Table 4.3: The best hyperparameters for predicting expert KTAS for each model

Additionally, a feature importance analysis using SHAP is conducted on the best performing model to get insights into the key variables that influence the KTAS predictions.

4.1.1 Decision Tree

The DT model demonstrated varying levels of performance when predicting registered nurse KTAS and expert KTAS, reflecting differences in complexity and consistency between the two classification tasks. For registered nurse KTAS (Table 4.2), the best performing model used the Gini criterion, a maximum tree depth of 20, minimum samples per leaf of 1, and minimum samples per split of 30. In contrast, the DT model that predicted expert KTAS levels (Table 4.3) had a slightly shallower maximum depth of 15 while maintaining the same values for the Gini criterion, minimum leaf samples, and minimum split samples. The deeper tree for registered nurse KTAS suggests that additional depth was beneficial in capturing the nuances of registered nurse-assigned triage decisions, whereas a slightly

shallower tree was sufficient for expert KTAS.

Furthermore, as can be seen in Table 4.1, the DT model performed better in predicting expert KTAS than in predicting registered nurse KTAS. The accuracy for expert KTAS was 65.51%, compared to 61.08% for registered nurse KTAS. Also, Cohen's Kappa for expert KTAS (0.4846) was also higher than for registered nurse KTAS (0.4403), highlighting better agreement between predictions and expert labels beyond random chance. The Weighted Kappa, which accounts for the ordinal nature of KTAS levels, was higher for expert KTAS (0.6328 vs. 0.5091) as well, reflecting in what extent the model is able to penalize large misclassifications. Additionally, the AUC for expert KTAS (0.7963) also surpassed that for registered nurse KTAS (0.7730). These results indicate that expert KTAS prediction was easier and discriminated better between urgency levels.

Table 4.4: Classification Report for Registered Nurse KTAS Predictions by DT

Class	Precision	Recall	F1 Score	Support
1	1.00	0.67	0.80	3
2	0.61	0.55	0.58	55
3	0.62	0.60	0.61	114
4	0.67	0.71	0.69	117
5	0.30	0.37	0.33	27
Weighted Avg	0.62	0.61	0.61	316

Table 4.5: Classification Report for Expert KTAS Predictions by DT

Class	Precision	Recall	F1 Score	Support
1	1.00	0.67	0.80	3
2	0.65	0.43	0.51	55
3	0.62	0.72	0.67	114
4	0.73	0.75	0.74	117
5	0.41	0.39	0.40	27
Weighted Avg	0.66	0.66	0.65	316

Tables 4.4 and 4.5 show that the F1 scores for registered nurse KTAS and expert KTAS predictions reveal notable differences in the model's ability to balance precision and recall across the five urgency levels. For registered nurse KTAS, the weighted average F1 score was 0.61, with a stronger performance in predicting KTAS 1 (F1 = 0.80) and KTAS 4 (F1 = 0.69), but considerably lower performance for KTAS 2 (F1 = 0.58) and KTAS 5 (F1 = 0.33). This disparity suggests that, even though registered nurse KTAS Level 1 has only three data points, the features associated with it are highly distinctive, enabling the model to effectively differentiate these cases. Conversely, the lower performance for KTAS 2 and KTAS 5 highlights potential feature overlap and challenges in distinguishing these levels from adjacent classes, emphasizing the complexity of accurately predicting these levels. The F1 scores for expert KTAS predictions were generally higher, with a weighted average of 0.65 and improved F1 scores for KTAS 2, 3, 4, and 5. The

higher weighted F1 score for expert KTAS indicates better overall balance in predicting both frequent and less frequent classes.

4.1.2 Random Forest

The best configuration for the RF model when predicting registered nurse KTAS included the Gini criterion, 250 estimators, a maximum tree depth of 20, minimum samples per leaf of 1, and minimum samples per split of 5 (Table 4.2). In contrast, the expert KTAS model performed best with 500 estimators, a shallower maximum depth of 15, minimum samples per leaf of 1, and a minimum split of 2 (Table 4.3). The increase in estimators and lower split threshold for expert KTAS suggests that the model benefited from more ensemble averaging and finer decision boundaries.

The RF model demonstrated improved performance compared to the DT model in predicting both registered nurse KTAS and expert KTAS, benefiting from its ensemble-based approach. This can be seen in Table 4.1. For example, the RF model achieved an accuracy of 64.87% when predicting registered nurse KTAS and 69.94% for expert KTAS, reflecting better overall classification compared to the DT model.

Furthermore, the Cohen's Kappa value when predicting registered nurse KTAS was 0.4680, while for expert KTAS it increased to 0.5448. This indicates that the RF model can more easily approximate expert decision making, possibly due to reduced variability in expert labeling. The Weighted Kappa showed similar trends, increasing from 0.6020 for registered nurse KTAS to 0.6538 for expert KTAS. This improvement shows that the RF can better account for the ordinal nature of the KTAS scale than the DT model. Additionally, with an AUC value of 0.5 indicating random guessing, the AUC for registered nurse KTAS (0.8799) and expert KTAS (0.8941) emphasize the RF's moderate capability in ranking and discriminating between urgency levels.

Table 4.6: Classification Report for Registered Nurse KTAS Predictions by RF

Class	Precision	Recall	F1 Score	Support
1	1.00	0.67	0.80	3
2	0.79	0.42	0.55	55
3	0.61	0.70	0.65	114
4	0.66	0.82	0.73	117
5	0.57	0.15	0.24	27
Weighted Avg	0.66	0.65	0.63	316

For RF, the F1 scores followed the same trend as observed for DT (Figures 4.6 and 4.7). For registered nurse KTAS, the weighted average F1 score was 0.63, with the highest performance observed for KTAS 1 (F1 = 0.80) and KTAS 4 (F1 = 0.73). However, the model struggled with KTAS 5 (F1 = 0.24), a recurring challenge. In contrast, the weighted average F1 score for expert KTAS was higher at 0.69, with improvements for all KTAS levels. Notably, KTAS level 1 obtained an F1 score of 1.00, meaning the model achieved perfect precision and recall for this class, likely due to the distinctiveness and severity of KTAS level 1 cases.

Table 4.7: Classification Report for Expert KTAS Predictions by RF

Class	Precision	Recall	F1 Score	Support
1	1.00	1.00	1.00	3
2	0.88	0.46	0.60	61
3	0.66	0.77	0.71	124
4	0.70	0.79	0.74	110
5	0.64	0.39	0.48	18
Weighted Avg	0.72	0.70	0.69	316

4.1.3 XGBoost

The XGBoost model performed approximately equal to the RF model in the prediction of expert KTAS and registered nurse KTAS (Table 4.1). In the prediction of registered nurse KTAS, the accuracy of XGBoost was 63.61%, which slightly outperformed the DT model (61.08%) but was lower than RF (64.87%). The weighted average F1 score was 0.62, which outperformed the DT model (0.61) but was lower than RF (0.63).

For expert KTAS predictions, XGBoost achieved an accuracy of 69.30%, which is higher than the accuracy of the DT model (65.51%) and approaches the RF model (69.94%). The weighted average F1 score was 0.69, on par with RF (0.69) and higher than DT (0.65). Cohen's Kappa values further highlight the comparative performance of XGBoost. For registered nurse KTAS predictions, Cohen's Kappa was 0.4550, higher than DT (0.4403) but slightly lower than RF (0.4680). The Weighted Kappa for registered nurse KTAS was 0.5850, again surpassing DT (0.5091) and again falling short of RF (0.6020). For expert KTAS predictions, Cohen's Kappa increased to 0.5410, close to RF (0.5448) and significantly higher than DT (0.4846). The Weighted Kappa for expert KTAS predictions was 0.6884, the highest among the models.

For AUC, XGBoost achieved an AUC value of 0.8444 for registered nurse KTAS and 0.8819 for expert KTAS. These results were better than DT (0.7730 and 0.7963) but slightly lower than RF (0.8799 and 0.8941).

Tables 4.8 and 4.9 display the classification reports for registered nurse KTAS and expert KTAS predictions, respectively. The XGBoost model resulted in the strongest performance for KTAS 1 (F1 = 0.67) and KTAS 4 (F1 = 0.72), but it struggled with KTAS 5 (F1 = 0.27). This seems to be a consistent challenge across all models. However, KTAS 1 predictions showed a significant lower F1 score (0.57) compared to RF (1.00). This is a crucial finding because KTAS 1 represents the most urgent cases, where accurate classification is essential.

This discrepancy in KTAS 1 predictions may be attributed to the unconventional hyperparameters identified for XGBoost during tuning. Notably, the best-performing configuration for expert KTAS predictions included a gamma value of 0, a learning rate of 1, a maximum tree depth of only 1, a minimum child weight of 3, and just 50 estimators (Table 4.3). These settings contrast with typical XGBoost expectations, which often favor deeper trees, more estimators, and lower learning rates to capture complex patterns effectively [65]. Instead, this configura-

tion suggests that the model prioritized simplicity, relying on shallow trees and a limited number of estimators to prevent overfitting. This may have led to a loss of critical decision making depth, impacting the model's ability to distinguish KTAS 1 patients effectively.

Table 4.8: Classification Report for Registered Nurse KTAS Predictions by XGBoost

Class	Precision	Recall	F1 Score	Support
1	0.67	0.67	0.67	3
2	0.69	0.44	0.53	55
3	0.61	0.68	0.64	114
4	0.66	0.79	0.72	117
5	0.50	0.19	0.27	27
Weighted Avg	0.63	0.64	0.62	316

Table 4.9: Classification Report for Expert KTAS Predictions by XGBoost

Class	Precision	Recall	F1 Score	Support
1	0.50	0.67	0.57	3
2	0.79	0.49	0.61	61
3	0.70	0.76	0.73	124
4	0.69	0.78	0.73	110
5	0.50	0.39	0.44	18
Weighted Avg	0.70	0.69	0.69	316

Given these findings, we chose to apply SHAP analysis to the RF model instead of XGBoost for three key reasons. First, RF outperformed, although slightly, XGBoost across all evaluation metrics with the only exception of Weighted Kappa in expert KTAS prediction, where XGBoost had a slight advantage. Second, RF demonstrated superior performance in identifying KTAS 1 cases, achieving an F1 score of 1.00, where XGBoost only achieved an F1 score of 0.57 for these most urgent cases. Third, XGBoost's poor KTAS 1 performance may stem from its unusually optimal hyperparameter settings and its overly simplistic structure limits its ability to capture complex patterns, making SHAP analysis on it less meaningful.

4.2 Results of SHAP Analysis

This section presents the results of the SHAP analysis conducted on the best performing model, the RF model, in predicting expert-assigned KTAS levels. The SHAP analysis provides insights into the key predictive variables influencing each KTAS level and allows an interpretation of the decision making process of the model. The complete rankings of the results of applying SHAP on the RF model can be found in Appendix D. These results should be interpreted as follows. When

looking at for example the results on KTAS level 1, each point represents a patient, and features are ranked by importance. The horizontal spread indicates the magnitude of a feature's impact on the model's prediction, while color shows the feature value (red = high, blue = low). For example, mental status (Mental_Unresponsive) and low saturation are among the top predictors, with lower saturation values (blue dots to the right) strongly pushing predictions toward KTAS level 1. We will now look at the results of our SHAP analysis for each KTAS level individually.

For KTAS level 1, the results indicate a strong association with markers of severe physiological instability and critical medical conditions. Patients assigned to KTAS 1 frequently present with begin unresponsive, low oxygen saturation, and hypotension, as reflected in reduced systolic and diastolic blood pressure. An increased heart rate further positively influenced KTAS 1 prediction. Also, some chief complaints, such as psychiatric or social emergencies, cardiac arrest, and respiratory disorders, are often assigned to KTAS 1. The mode of arrival is also a significant variable, with patients arrived by private or public ambulance having a greater chance of being classified as KTAS 1, and those arrived by private car having less chance of being classified in this category. Furthermore, older patient age and lower NRS scores have high influence on level 1 prediction, and low patient arrivals per hour appear to increase the likelihood of the KTAS 1 classification. Surprisingly, female patients reveal a slightly raised tendency for sorting into the most severe triage category compared with male patients.

For KTAS level 2, SHAP analysis indicates complaints such as chest pain, psychiatric or social emergencies, weakness, and hemorrhage to have a positive influence on this KTAS level. Extremely low or very high pain intensity significantly affects KTAS 2 classification as well. Patients who are older and have a lower heart rate are also shown to be classified under this level. Moreover, mode of arrival influences KTAS 2 assignment, with private vehicle arrival having a negative correlation with this level and private ambulance arrival having a positive correlation. Unresponsive and fully alert patients both have a negative effect on KTAS 2 classification. In addition, abdominal pain negatively impacts KTAS 2 classification. Contrary to KTAS 1 findings, male patients are classified more frequently at this level than female patients.

Patients classified as KTAS level 3 exhibit moderate urgency, requiring timely but not immediate intervention. SHAP analysis reveals pain level, either extremely low or high, as the most influential variable for triage as KTAS 3. No pain and no injury are positively correlated with this level as well. Some complaints, including abdominal pain, dizziness, hemorrhage, and neurological symptoms, strongly influence KTAS 3 triage. Besides, elevated systolic and diastolic blood pressure are also positive indicators. Signs of mental status show that KTAS 3 patients are generally alert and responsive, as opposed to the more impaired mental status for KTAS 1 and 2. Physiological measures, such as oxygen saturation and respiratory rate, remain within elevated parameters. Also, female patients seem to get classified more frequently in KTAS 3 than male patients. Additionally, the absence of certain complaints, such as chest pain and psychiatric illness, distinguishes KTAS 3 from more acute levels, while the absence of skin complaints, peripheral

vascular/leg pain, and procedural needs distinguishes it from less acute levels.

For KTAS level 4, the most important feature revealed by SHAP values is the binary marker of pain, where patients arriving with pain have a positive influence on the prediction of this KTAS level. The NRS score for pain, which resulted in the second highest importance, shows that patients with moderate pain are likely to fall under KTAS 4. Secondly, the presence of injuries, particularly laceration injuries and trauma injuries, is a deciding predictor. Physiologic signs such as increased oxygen saturation, normal to high systolic and diastolic blood pressure, reduced respiratory rates, and increased blood temperature also correlate in a positive way with KTAS 4 classification. The absence of complaints such as chest pain, dizziness, psychiatric disorder, and dyspnea further distinguishes KTAS 4 from other triage categories. Mode of arrival is another key factor, where patients arriving by public or private ambulance are less likely to be assigned to this level, while patients arriving by private vehicle have a greater change to be assigned to KTAS 4.

KTAS level 5 are the least urgent cases. The most predictive factors for this category are the absence of pain and the presence of complaint categories such as skin-related issues, peripheral vascular/leg pain, abdominal problems, medication refills, eye-related concerns, and routine follow-ups. These patients are typically those who walk in or arrive by private vehicle, which is consistent with the fact that their conditions are non-emergency. Physiological indicators of high oxygen saturation and low body temperature are also associated with KTAS 5 classification. SHAP analysis also shows that younger patients contribute to classification at this non-urgent triage level.

4.3 Results of GPT-4o

This section presents the findings from GPT-4o's KTAS classification. First, we will focus on the triage rules extracted from the GPT-4o model. Second, we will show the results of applying various prompting techniques on GPT-4o, including these extracted triage rules. We interpret and discuss the results in Section 5.1.2.

4.3.1 Triage Rule Extraction

To assess whether GPT-4o can generate triage rules, we prompted the model to output IF-THEN decision rules for KTAS classification. The extracted rules define specific thresholds for specific variables, aiming to categorize patients into KTAS levels from 1 to 5.

In Listing 4.1, we present the complete set of GPT-4o-extracted rules for each KTAS level. While important features for each KTAS level according to GPT-4o are displayed in the listing, we want to highlight that the logical structure of the rules for KTAS Levels 1 through 3 primarily relied on `OR` conditions. In contrast, the rules for KTAS Levels 4 and 5 were characterized by multiple `AND` conditions. A discussion on these results can be read in Section 5.1.2.

Since it remains unclear how well these rules align with expert KTAS guidelines, we further investigated this by incorporating them into a secondary prompt

to GPT-4o. This way, we can assess whether GPT-4o could improve its own consistency when following its own self-derived rules. This evaluation is detailed in the following section, Section 4.3.2.

```
# Rules for KTAS Level 1:
IF Mental == Unresponsive OR (Complaint_category == Cardiac Arrest
    ) OR (Saturation < 85) OR (HR > 150 OR HR < 40) OR (SBP < 80)
    OR (RR > 35 OR RR < 8) OR (BT > 41 OR BT < 32) THEN 1

# Rules for KTAS Level 2:
IF (Mental == Verbal Response) OR (Complaint_category == Chest
    Pain AND (Saturation < 90 OR HR > 130 OR SBP < 90)) OR (
    Complaint_category == Convulsions; Seizures) OR (NRS_pain >= 8)
     OR (BT >= 39 AND BT < 41) OR (Saturation >= 85 AND Saturation
    < 90) THEN 2

# Rules for KTAS Level 3:
IF (Mental == Pain Response) OR (Complaint_category == Abdominal
    Pain AND NRS_pain >= 6) OR (Complaint_category == Respiratory
    Problems AND (Saturation >= 90 AND Saturation < 94)) OR (HR >=
    110 AND HR <= 130) OR (SBP >= 90 AND SBP < 100) OR (BT >= 38
    AND BT < 39) OR (NRS_pain >= 4 AND NRS_pain < 8) THEN 3

# Rules for KTAS Level 4:
IF (Mental == Alert) AND (Complaint_category == Back Pain OR
    Complaint_category == Ear/Nose/Throat Problems OR
    Complaint_category == Dental; Toothache OR Complaint_category
    == Skin Complaint/Trauma) AND (NRS_pain >= 2 AND NRS_pain < 4)
    AND (SBP >= 100 AND SBP < 120) AND (HR >= 60 AND HR < 110) AND
    (RR >= 12 AND RR < 20) AND (BT >= 36 AND BT < 38) THEN 4

# Rules for KTAS Level 5:
IF (Mental == Alert) AND (Complaint_category == Follow-Up OR
    Complaint_category == Medication Refill OR Complaint_category
    == Unknown Problem) AND (NRS_pain < 2) AND (SBP >= 120 AND SBP
    <= 140) AND (HR >= 60 AND HR <= 100) AND (RR >= 12 AND RR <=
    20) AND (BT >= 36 AND BT <= 37.5) THEN 5
```

Listing 4.1: GPT-4o's Extracted Rules for each KTAS Level

4.3.2 Prompt Engineering Techniques

This section shows the results of various prompt engineering techniques that have been tested to improve the classification performance of GPT-4o in KTAS level prediction. All prompting techniques were evaluated against the expert-assigned KTAS labels, which serve as the gold standard for this study. The evaluation metrics are accuracy, weighted F1-score, Cohen's Kappa, Weighted Kappa, and AUC. The Weighted Kappa metric is emphasized, as it measures agreement between GPT-4o and expert-assigned KTAS levels considering ordinal classification errors. The results are categorized into four groups: (1) standard prompting techniques, (2) SHAP-informed prompting techniques, (3) experimental prompting techniques,

and (4) a prompt with the triage rules extracted from the GPT-4o model itself.

Table 4.10 displays the results of these experiments, with the zero-shot prompt as the baseline. A performance measure that is higher than the corresponding value of the zero-shot prompt is highlighted in bold. In addition, prompting techniques that outperform the baseline across all five metrics are fully bolded for emphasis.

To investigate the classification behavior across the KTAS levels, Table 4.11 presents the F1 scores per KTAS level. The bold values represent the KTAS level with the highest F1 score for each prompting technique.

Prompting Technique	**Accuracy**	**F1-Score**	**Cohen's Kappa**	**Weighted Kappa**	**AUC**
Zero-shot (baseline)	*0.4873*	*0.4799*	*0.2350*	*0.4374*	*0.6126*
Few-shot	**0.5063**	**0.4960**	**0.2572**	0.4136	**0.6222**
Role	0.4810	0.4724	0.2249	0.4315	0.6078
Implicit-CoT	**0.5032**	**0.4972**	**0.2662**	**0.4515**	**0.6303**
Fluent style	**0.5000**	**0.4946**	**0.2633**	**0.4722**	**0.6282**
SHAP-informed	**0.5411**	**0.5628**	**0.3722**	**0.5552**	**0.6956**
SHAP + exceptions	**0.5728**	**0.5874**	**0.4015**	**0.5526**	**0.7072**
SHAP + exceptions + implicit-CoT + fluent	**0.6044**	**0.6100**	**0.4302**	**0.6186**	**0.7172**
Emotional-based	0.4715	0.4655	0.2174	0.4264	0.6041
Reward-based	0.4652	0.4587	0.2061	0.4171	0.5982
Nonsensical-based	**0.4905**	0.4797	0.2341	0.4269	0.6125
Constraint - use everything but complaint	**0.5032**	**0.4915**	**0.2556**	**0.4704**	**0.6244**
Constraint - use only complaint	**0.5222**	**0.5140**	**0.2912**	**0.5352**	**0.6415**
Time constraint	**0.5000**	**0.4939**	**0.2616**	**0.4851**	**0.6272**
Math puzzle	**0.4968**	**0.4900**	**0.2543**	**0.5173**	**0.6227**
No prior knowledge	**0.5000**	**0.4932**	**0.2629**	**0.4853**	**0.6267**
Adversarial information	0.4778	0.4663	0.2127	**0.4565**	0.6017
Extracted rules by GPT-4o	0.4873	0.4756	**0.2491**	0.2841	**0.6277**

Table 4.10: Performance Prompting Techniques

Prompting Technique	**KTAS 1**	**KTAS 2**	**KTAS 3**	**KTAS 4**	**KTAS 5**
Zero-shot (baseline)	***0.6667***	*0.3964*	*0.4981*	*0.5398*	*0.2400*
Few shot	**0.7500**	0.4696	0.5401	0.5023	0.2000
Role	**0.7500**	0.3738	0.4844	0.5470	0.2222
Implicit-CoT	**0.6667**	0.4426	0.5323	0.5403	0.1481
Fluent style	0.5455	0.4394	0.5038	**0.5631**	0.1905
SHAP-informed	0.4615	0.5932	**0.6320**	0.5208	0.2564
SHAP + exceptions	0.5455	0.6050	**0.6356**	0.5784	0.2581
SHAP + exceptions + implicit-CoT + fluent	0.6000	0.6261	**0.6586**	0.6038	0.2609
Emotional-based	**0.5455**	0.3932	0.4724	0.5310	0.2500
Reward-based	**0.6000**	0.3793	0.4646	0.5263	0.2500
Nonsensical-based	**0.7500**	0.3818	0.5132	0.5374	0.1818
Constraint - use everything but complaint	0.5455	0.4800	0.5481	**0.5463**	0.1905
Constraint - use only complaint	**0.7500**	0.3604	0.5019	0.5974	0.1739
Time constraint	**0.6000**	0.4409	0.5057	0.5592	0.1739
Math puzzle	**0.6667**	0.4553	0.5095	0.5352	0.1667
No prior knowledge	0.5455	0.4769	0.4961	**0.5472**	0.1905
Adversarial information	**0.7500**	0.3590	0.5000	0.5392	0.1053
Extracted rules by GPT-4o	**0.6667**	0.1579	0.5705	0.5914	0.1587

Table 4.11: F1 Scores across all Prompting Techniques for each KTAS Level

4.4.2.1 Standard Prompting

The zero-shot prompt achieved a Weighted Kappa of 0.4374, which we used as a benchmark for later comparisons. Few-shot prompting improved in most metrics

compared to zero-shot, although its Weighted Kappa (0.4136) was slightly lower. Role prompting obtained a Weighted Kappa of 0.4315, similar to zero-shot but lower than Implicit-CoT and fluent style prompting, that resulted in Weighted Kappa values of 0.4515 and 0.4722, respectively. Both techniques outperformed zero-shot prompting in all performance metrics.

When looking at the performance for each KTAS level (Table 4.11), we see that zero-shot prompting, role prompting, and implicit-CoT prompting all achieved the highest F1-score for KTAS level 1 (0.6667, 0.7500, 0.7500, 0.667, respectively). However, fluent style prompting achieved a slightly higher F1 score for KTAS level 4 (0.5631). All prompting techniques performed the worst when predicting KTAS level 5.

4.4.2.2 SHAP-Informed Prompting

The prompts in this group explicitly incorporates feature importance derived from SHAP applied to the best performing classical AI model, the RF.

Basic SHAP-informed prompting achieved a Weighted Kappa of 0.5552, an improvement over the best performing standard prompting method (fluent style at 0.4722). Allowing GPT-4o to make exceptions to the SHAP-informed prompt further boosted performance. This prompt resulted in a Weighted Kappa of 0.5526, with a notable AUC of 0.7072, the highest observed thus far.

The most comprehensive method turned out to be SHAP-informed + Exceptions + Implicit-CoT + Fluent. With a Weighted Kappa of 0.6186, it outperformed all other methods, showing that combining data-driven insights from SHAP, allowing for exceptions, suggesting to reason step-by-step before classifying, and giving patient data in a fluent style results in the best predictions. The AUC of 0.7172 further corroborates the robustness of this approach compared to the other prompting techniques.

Table 4.11 shows that all SHAP-informed prompting techniques achieved the highest F1 scores (0.6320, 0.6356, and 0.6586) for KTAS level 3 and the lowest F1 scores (0.2564, 0.2581, and 0.2609) for KTAS level 5.

4.4.2.3 Experimental Prompting

The techniques in this approach were designed to explore how alternative framing strategies influence the classification performance of GPT-4o.

Emotional-based, reward-based, and nonsensical-based prompting resulted in lower performance across all metrics compared to zero-shot prompting. However, providing GPT-4o with adversarial information by misleadingly stating that a KTAS value of 1 indicates the lowest urgency and 5 the highest, while the opposite is true, unexpectedly led to a higher Weighted Kappa than zero-shot prompting. However, all other performance metrics showed lower scores, although the differences were not significant.

In contrast, all other prompting techniques in this group, except for the adversarial information prompt, outperformed zero-shot prompting across all evaluation metrics. For the adversarial information prompt, in which we prompted that KTAS 1 corresponds to the least urgent level and KTAS 5 to the most urgent level, while

the opposite is true, surprisingly led to an increased Weighted Kappa compared to zero-shot. These outcomes are both intriguing and unexpected, which we will further discuss in Section 5.1.2.

When looking at the F1 scores for the experimental prompting techniques (Table 4.11), we see that all prompting techniques, except the prompt with the constraint to use every variable except for the complaint and the no prior knowledge prompt, which achieved the highest F1 scores for KTAS level 4, obtained the highest F1 scores for KTAS level 1. Again, all prompting techniques obtained the lowest F1 score for KTAS level 5.

4.4.2.4 Self-Extracted Rule Prompting

When prompted with its own extracted triage rules, GPT-4o achieved a Weighted Kappa of 0.2841, which was lower than the zero-shot baseline (0.4374) and all other evaluated prompting techniques. The accuracy (0.4873), F1-score (0.4756), and Cohen's Kappa (0.2491) were also lower than most other methods. However, the AUC (0.6277) was slightly higher than the zero-shot baseline (0.6126). A detailed discussion of these results is provided in Section 5.1.2.

Despite its lower overall performance, the extracted rule prompt yielded moderately strong F1 scores (Table 4.11) for KTAS levels 3 (0.5705) and 4 (0.5914). This prompting technique performed particularly bad when predicting KTAS levels 2 and 5 (0.1579 and 0.1587, respectively).

5 Discussion

In this discussion, we examine the performance of classical ML models and GPT-4o in KTAS classification. We compare DT, RF, and XGBoost predictions for registered nurses and experts, and analyse feature importances using SHAP on our RF model. We review triage rules generated by GPT-4o, comparing them with SHAP-based insights, and assess the impact of different prompt engineering techniques on GPT-4o's classification performance. Additionally, we compare our research with previous research and discuss the ethical implications of using LLMs for triaging patients.

5.1 Interpretation of Results

This section discusses the results of DT, RF, and XGBoost when predicting KTAS levels assigned by both registered nurses and experts as well as the results of GPT-4o's extracted triage rules and the impact of different prompt engineering techniques on its KTAS classification performance.

5.1.1 Discussion Classical ML Models

First, we examine the performance of DT, RF, and XGBoost in predicting either registered nurse KTAS or expert KTAS. We will also discuss the findings of applying SHAP analysis on our best performing classical ML model.

5.1.1.1 Performance Classical ML Models

The performance analysis of our classical ML models shows that while XGBoost achieves a better Weighted Kappa value when predicting expert KTAS labels, RF achieves the highest results in both KTAS classification tasks for every other prediction task. These findings slightly contradict our initial hypothesis, as we expected XGBoost to outperform both RF and DT due to its ability to handle imbalanced datasets and capture complex feature interactions [78].

This contradiction with our hypothesis may be explained by the hyperparameters that optimized the performance of XGBoost for expert KTAS. We found that optimal performance was reached with a maximum tree depth of 1, a learning rate of 1, and just 50 estimators. This is unusual, as XGBoost generally relies on deeper trees and lower learning rates to improve its performance [65]. That such a shallow tree structure performed optimally suggests that expert KTAS levels have a relatively simple decision boundary that can be effectively modeled by

weak learners (decision stumps). However, that also suggests XGBoost's decision making process is shallow and does not rely on feature interactions as much as RF. Because SHAP analysis is applied to account for model prediction making, applying it to XGBoost in this case would yield limited insights, as the model exploits simple decision rules and not intricate dependency among features.

In comparison, RF needed deeper trees and more estimators to obtain optimal performance. This shows that RF is able to capture more complex relationships between variables compared to the shallow tree from XGBoost, making it a better choice for SHAP analysis, as it gives a more comprehensive and insightful understanding of feature importance. However, we must note that while RF achieved a perfect F1 score for KTAS 1, these results are based on a very small support size of $n = 3$ (Table 4.7). With such a small support size, the results should be interpreted with caution, as the high performance could partly be due to random chance rather than true predictive power.

Interestingly, every model faces challenges when it comes to predicting KTAS 5 in both classification tasks. The F1 scores for KTAS 5 are consistently the lowest, especially for registered nurse KTAS (DT: 0.33, RF: 0.24, XGBoost: 0.27). This could be attributed to the limited number of KTAS 5 cases in the dataset, and therefore also in the training set, which makes it difficult for the models to effectively differentiate these cases. Moreover, KTAS 5 cases may exhibit feature overlap with KTAS 4, further complicating classification.

To continue, in all the models we have examined, expert KTAS predictions outperform those of registered nurse KTAS, confirming our hypothesis that decisions made by experts are more consistent and thus easier for ML models to learn. This could be because expert clinicians have a tendency to employ more consistent and standard triage criteria, which would lead to a more formalized decision making process. On the other hand, registered nurses can use more subjective judgment and variation in their assessments, and therefore their assignments to KTAS will be less deterministic for the ML models. This is in line with previous research that suggests clinical expertise reduces variation in decision making [176], explaining why our classical ML models may have performed better in KTAS expert classification than in registered nurse classification.

These results highlight some key points about using ML models in ED triage. While the Weighted Kappa values for RF and XGBoost show a fair amount of agreement with expert assessments, they are not high enough to rely on these models completely in real-life emergency settings. In these high-stakes clinical environments, even minor classification errors can have significant consequences. The comparatively modest Weighted Kappa scores, even for the best performing RF model, indicate that ML models remain far from expert-level performance and require additional improvements to use them in actual triage practice.

To conclude, RF turned out to be the best performing model in KTAS assignment prediction, closely followed by XGBoost. DT fails to capture the intricateness of the triage decision making process as good as RF and XGBoost. RF was used for SHAP analysis because it performed better than the other models in most evaluation metrics and was able to capture deeper feature interactions that ensure greater interpretability of the results. The differences between expert and

registered nurse KTAS predictions suggest that the expert classifications are more structured and thus easier to model. Future research must examine how to improve predictions for minority classes, particularly KTAS 5.

5.1.1.2 Feature Importance Analysis of Best Performing Classical ML Model

SHAP analysis provides an explanation of the driving forces behind KTAS classification, revealing expected and unexpected correlations within the data. The implications of these findings are discussed here, comparing their alignment with our expectations.

One notable discovery is the gradual change in physiological instability as we move from KTAS 1 to KTAS 5. For KTAS 1, severe abnormalities in vital signs, unresponsiveness, low blood pressure, low oxygen levels, and rapid heart rate, are strong indicators for classification, reinforcing the idea that extreme physiological distress is a crucial factor for immediate intervention. On the other hand, KTAS 5 patients show stable physiological signs, characterized by high oxygen saturation, low body temperature, and no urgent complaints. This aligns with common clinical knowledge, as triage systems are meant to prioritize life-threatening conditions based on visible physiological changes.

However, KTAS 2 and KTAS 3 present intriguing overlaps and divergences. While KTAS 2 is characterized by urgent conditions like chest pain and hemorrhage, it does not consistently feature severe physiological instability. This suggests that while life-threatening conditions are likely, they are often in an early or moderate state of progression. KTAS 3, on the other hand, exhibits a broad spectrum of conditions with moderate urgency, where pain (either extremely high or low) serves as a dominant predictor. The role of blood pressure is particularly notable here, as hypertension becomes more relevant at this level. This shift from the acute instability seen in KTAS 1-2 to a more moderate instability in KTAS 3 highlights a more sophisticated classification system.

Pain emerges as a complex variable across KTAS levels. Patients who are classified as KTAS 1 typically have lower NRS pain scores, possibly due to unconsciousness or severe physiologic compromise precluding subjective pain perception. In contrast, KTAS 4 is positively influenced by the presence of pain, particularly moderate pain. KTAS 5, on the other hand, is strongly associated with the absence of pain. The complexity of interpreting the pain scale variable accurately is also highlighted by Moon et al. [101], showing that pain alone is not always a reliable determinant of acuity and can contribute to triage misclassification when assessed in isolation. This may suggest that pain serves as a secondary rather than a primary determinant in triage classification, with its influence being context-dependent rather than absolute.

To continue, the mode of arrival serves as an implicit indicator of patient severity, which follows an expected pattern in triage. Patients who need urgent medical care are typically arriving via private or public ambulance, as their situations often require immediate attention. Moreover, those who are seriously ill or injured usually do not have the strength or time to walk to the ED or arrange for a ride.

On the other hand, patients with less severe conditions are more likely to arrive by private vehicle or even on foot, indicating that their health is more stable and they do not need urgent care.

Furthermore, our SHAP analysis showed that female patients were more likely to be classified as KTAS 1 and 3, while male patients were more likely to be classified as KTAS 2. Although these differences may reflect actual patterns in our dataset, we should still be cautious. Gender-based variation in triage classification could indicate underlying bias in how decisions are made, either by the model or in the original labeling. Since triage levels directly impact patient treatment and priority, even small gender-related differences should be carefully examined to ensure decisions are fair and clinically justified.

Another insight from the SHAP analysis is that a lower number of patient arrivals at the same hour as the patient itself arrived was linked to a higher chance of receiving a Level 1 classification. This indicates that when the ED was less crowded, with just a few patients showing up, those patients were more likely to be triaged as Level 1. This was also stated in the Emergency Severity Index Handbook [100], that showed that triage decisions can be inappropriately used due to the capacity and bed availability of the ED.

To sum up, the SHAP analysis largely aligns with what we hypothesized. It underscores the strong links between higher triage levels and urgent complaint categories, including psychiatric or social issues, cardiac arrest, chest pain, unresponsive mental states, and patients arriving by either private or public ambulance. Conversely, lower levels of urgency were typically found among alert patients who arrived by private transport or on foot and with less severe complaints such as follow-ups, procedures, and normal physiological results. Interestingly, our SHAP analysis revealed that female patients were more associated with KTAS 1 and 3, while male patients were more associated with KTAS 2. This highlights the need for further investigation into potential genderbias in triage classification. Additionally, the complex pattern of pain levels indicates that pain is not a straightforward measure of medical urgency and should be interpreted together with other factors such as chief complaints and vital signs. Furthermore, the analysis showed that a lower number of patient arrivals per hour was a positive predictor for KTAS 1 classification, implying that during less busy periods, triage staff may have more capacity to accurately identify and classify critically ill patients.

5.1.2 Discussion GPT-4o

Next, we will discuss the behavior of GPT-4o under different prompt engineering strategies for the prediction of KTAS levels. We will first examine the triage rules extracted from GPT-4o and then continue to explore the various prompt engineering strategies.

5.1.2.1 Extracted Rules from GPT-4o

Both GPT-4o and SHAP have showed variables such as unresponsiveness, cardiac arrest, low saturation, and low SBP to be positive indicators for KTAS level 1. This

strongly aligns with established triage systems, and thus our hypothesis, that both methods would identify critical factors as immediate life threats requiring urgent resuscitation. For example, the CTAS categorizes cardiac arrest, respiratory arrest, and altered levels of consciousness as Level 1 emergencies [99]. So, it is reassuring to see that both SHAP analysis and GPT-4o highlight these critical factors for KTAS Level 1.

These models also agree that chest pain is a significant feature for KTAS Level 2. In emergency triage, chest pain is indeed a classic high-priority complaint because it could represent life-threatening condition [99]. The SHAP analysis highlighted chest pain as a vital predictor for Level 2, and GPT-4o similarly included it in its guidelines for Level 2. This agreement emphasizes that both approaches acknowledge that a patient with chest pain should be considered urgent, which again supports our hypothesis.

A notable difference, however, is that the extracted triage rules from GPT-4o did not treat all chest pain equally, as it added extra conditions under which chest pain leads to Level 2. For instance, GPT-4o suggests that chest pain should occur in combination with a saturation lower than 90, a heart rate above 130, or a SBP-level lower than 90. This indicates that GPT-4o was trying to mirror the way human triage nurses use modifiers or context for chest pain. These nuances were also shown by the Ministry of Health and Long-Term Care [99], which classifies patients with chest pain in combination with severe hypoperfusion as CTAS Level 1, in combination with borderline perfusion as CTAS Level 2, while in combination with noncardiac features as CTAS Level 3. So, it seems that GPT-4o is potentially drawing on such medical patterns in which features could interact. Note that this might also have happened for RF predicting KTAS levels, but SHAP cannot visualize this.

Furthermore, as observed in the SHAP analysis, NRS pain had a complex relationship with KTAS predictions, suggesting that pain interacts with other features, such as specific chief complaints or physiological observations, to determine acuity levels. In contrast, GPT-4o's extracted rules simplified the role of pain, overlooking the non-straightforward pattern we observed in the SHAP analysis. It categorizes NRS pain scores of 8 or higher as KTAS 2, scores between 4 and 8 as KTAS 3, scores from 2 to 4 as KTAS 4, and anything below 2 as KTAS 5 (Listing 4.1). This strict approach likely leads to misclassifications when using these rules in its prompt. The results indeed showcase this: when the rules generated by GPT-4o were provided in the prompt, the Weighted Kappa dropped to 0.2841, which is significantly lower than the 0.4374 achieved with zero-shot prompting. This indicates that the extracted triage rules from GPT-4o do not adequately capture the complexities of pain involved in triage decisions.

It is unclear why GPT-4o oversimplifies NRS pain. Perhaps the prompt that was given to it to produce triage rules was not ideally suited to drawing out the complex, multifaceted decision criteria we need. Another reason might be that there can be an underlying limitation in GPT-4o's reasoning capacity in writing rules for considering multiple interacting features, even when it was specifically asked to do so. Or, GPT-4o simply may not have enough knowledge about how pain is covered by triage procedures and therefore could lead to a rather simplistic

perspective on its significance. This raises important concerns about the reliability of LLMs as research tools in clinical reasoning tasks. Therefore, GPT-4o's current capabilities may fall short for high-stakes medical reasoning in research applications.

These findings emphasize the importance of diving deeper into how GPT-4o processes and creates decision rules for complex classification tasks. Future research could investigate more refined prompt engineering strategies, or investigate if the model can be trained to identify multi-feature dependencies, and explore any knowledge gaps in LLM-based triage decision making. Identifying these limitations is important to improving AI-assisted triage systems and making sure they align with real-world clinical reasoning.

Another issue is that GPT-4o left out some significant variables identified by the SHAP analysis. Key variables such as arrival mode, patient age, injury presence, and DBP values were high predictors in the RF model, but were completely left out of the extracted rules from GPT-4o. This indicates that GPT-4o may be ignoring some less obvious or nonphysiological factors that are still highly important in triage decision making. While the prompt clearly stated that GPT-4o should consider these factors, they were not present in its parameters. Perhaps this is because the model is based on general medical facts, where, for example, ambulance transport and age are established as risk factors but are not generally considered first-order factors in assessing urgency. This shows us to be cautious when interpreting the rules produced by GPT, as they may not capture the full richness of real triage decision making.

Continuing, particularly in lower acuity triage levels, the patient's chief complaint category often plays a significant role according to SHAP analysis. The analysis shows that the RF model took into account a broad set of complaint types. For example, SHAP might show that complaints such as "medication refill," "back pain," or "procedure" all nudged predictions toward Level 4 or 5, aligning with our hypothesis. In contrast, GPT-4o's rules incorporated a much narrower subset of complaints, despite the prompt explicitly providing all possible complaint categories for rule generation.

While this is not per se failure, it shows that we should be cautious. The rules generated by GPT-4o could be too specific, omitting the broad range of real-world triage factors. We attempted to get GPT-4o to produce a minimum of twenty rules per each KTAS level and to account for all complaints in our dataset, but these attempts resulted in inconsistent or unreliable rules. For example, GPT-4o often assigned the same complaint to different KTAS levels without combinations with other variables. This discrepancy calls for a cautious validation when using GPT-4o in order to infer triage classification rules.

Furthermore, beyond content differences, there is a structural contrast in the way GPT-4o formulated its rules. For KTAS Levels 1 to 3, the rules often operated on `OR` logic. One example is the following rule, that assigns unresponsive patients to KTAS level 1: `IF Mental == Unresponsive THEN 1`. This indicates that if a patient is unresponsive, they automatically get classified as KTAS 1, no matter what other clinical factors are . However, the rules for Levels 4 and 5 heavily relied on multiple `AND` conditions. According to GPT-4o's rules, a patient

must be alert and have a complaint related to follow-up, medication refills, or an unknown issue, while also meeting specific criteria for NRS pain, SBP, HR, RR, and BT, to be classified as KTAS 5. This kind of conjunctive rule forces that all low-risk conditions are met. On the one hand, this might align well with clinical reasoning, as patients classified as KTAS 5 should have no significant red flags and only minor complaints, while low acuity is characterized by the absence of those features, often requiring all clinical parameters to fall within normal limits. On the other hand, this approach might cause problems if GPT-4o's rules are too inflexible. The model, for example, requires that a KTAS 5 patient to present with one of the complaints: follow-up, medication refill, or unknown problem. When using these rules in practice, this causes GPT-4o to mislabel patients with presentations that are positive indicators for KTAS 5 but do not have one of these tree complaints. This suggests that GPT-4o's rules may be too inflexible for real-world clinical use.

To conclude, the triage classification rules generated by GPT-4o revealed both alignments with and deviations from our hypothesis. For KTAS 1, both GPT-4o and SHAP analysis identified features such as unresponsiveness, cardiac arrest, low oxygen saturation, and hypotension. In the case of KTAS Level 2, they both recognized chest pain as a significant indicator, but GPT-4o also included additional conditions such as low saturation or a high heart rate, showing an attempt to capture more clinical nuances. However, GPT-4o oversimplified the role of NRS pain by strictly assigning thresholds to KTAS levels. Moreover, GPT-4o did not use some important predictors identified by SHAP, such as mode of arrival, age of patient, and presence of injury. It also used only a few chief complaints and therefore lacked some relevant categories. Structurally, GPT-4o preferred `OR` conditions for high-acuity and `AND` conditions for low-acuity classifications. This could lead to the misclassification of cases that did not strictly match the predefined criteria. In short, although GPT-4o showed a systematic method of rule inference, its uncertainties present challenges to applying it to clinical triage decision making.

5.1.2.2 Performance Prompt Engineering Techniques

In this section, we discuss the performance of using different prompt engineering techniques on GPT-4o. In addition to overall performance, we compared GPT-4o and the classical ML models at the per-class level using F1 scores across all KTAS levels.

Standard Prompting

Among the standard techniques, few-shot and role prompting showed similar or even slightly worse performance compared to zero-shot prompting. This indicates that GPT-4o understanding of emergency triage does not gain much from having more labeled examples or from the experience of a head emergency triage nurse of over 20 years, which contrasted with our hypothesis, as we expected that these prompting techniques would improve performance over zero-shot prompting. For few-shot prompting, this is possibly because we may not have provided sufficient

varied examples, or because we only provided two examples. In the case of role prompting, the reason that its performance is almost zero-shot may be because GPT-4o is potentially already optimally tuned to abide by well-documented medical standards. It seems like these techniques might just be adding redundant information that is not really necessary in the task of assigning patients an urgency level according to the KTAS guidelines.

In contrast, implicit-CoT prompting led to a modest improvement in performance, aligning with our expectations. This shows that when prompted to reason sequentially, the model seems to organize its internal reasoning in a way that approximates clinical diagnostic reasoning better than zero-shot prompting. Prompting with a fluent style for the patient information resulted in the highest performance in this group. The improvement is probably the result of the model better aligning its output with the linguistic and logical structure of clinical reasoning, making its responses more context aware and easier to map onto established triage criteria.

When looking at the results of the class-level F1 scores, standard prompting techniques performed reasonably well for KTAS level 1, with few-shot and role prompting both achieving an F1 score of 0.75. This suggests that GPT-4o is able to recognize high-acuity cases relatively well, which may be due to clear clinical patterns commonly associated with severe presentations. In addition, we see a similar trend for all standard prompting techniques: KTAS level 5 is the most difficult to predict, consistently yielding the lowest F1 scores. This indicates that lower-acuity cases may have more ambiguous clinical features or feature interactions or lack strong, distinctive patterns.

SHAP-Informed Prompting

SHAP-informed prompting showed some impressive results, particularly when combined with flexible rule exceptions, implicit-CoT reasoning, and fluent style input. This aligns with our hypothesis, as we were indeed expecting the LLM to benefit from additional information obtained from SHAP. These findings suggests that by providing GPT-4o with feature importance insights from the RF model, we can effectively help GPT-4o focus on the features that are clinically relevant, improving its alignment with expert triage classifications.

However, solely suggesting that GPT-4o may make exceptions to the results from SHAP did not significantly change performance compared to the baseline SHAP-informed prompt. This suggests that allowing GPT-4o some flexibility in overriding SHAP-derived suggestions neither significantly improves nor degrades accuracy. It is possible that the model was already capable of making reasonable exceptions based on case specifics without requiring explicit permission.

It turns out that the most effective prompt was one that combined SHAP-informed information, exceptions, an implicit-CoT approach, and a fluent writing style as input. This prompt achieved a Weighted Kappa of 0.6186, suggesting that GPT-4o's internal processes can be directed toward more accurate predictions when it has the right external context to work with.

When focusing on the F1 scores for each KTAS level, we see that the reason why SHAP-informed prompting achieved the best overall performance among all

strategies is because it maintains relatively strong F1 scores across all classes. SHAP-informed prompts performed well not only on KTAS level 1, but also on level 4 and particularly level 3, where other prompting techniques achieved lower F1 scores. This suggests that structured, model-derived guidance may be particularly useful for improving GPT-4o's performance in mid-acuity decisions.

However, we want to note that we applied SHAP on our best performing RF model. But this model achieved an accuracy of only 69.94% and a Weighted Kappa of only 0.6538 for expert KTAS prediction. So, this raises the following question: if SHAP analysis were performed on an ideal ML model that perfectly predicted expert KTAS labels, would providing the results of applying SHAP on that model improve GPT-4o's performance even further?

Experimental Prompting

Several experimental prompts were tried, most of which performed worse than the baseline. Emotional-based, reward-based, and nonsensical-based prompts all produced Weighted Kappa scores below 0.43, suggesting that introducing emotional support or behavioral conditioning does not improve GPT-4o's triage forecasts.

Interestingly, when the model was constraint to use only chief complaints, it actually performed quite well. It suggests that if we highlight chief complaints in the prompt, it might encourage GPT-4o to focus more on this variable, which could boost classification accuracy. However, when we prompted to use all variables except for the chief complaint also resulted to better performance compared to zero-shot prompting, raising questions about whether GPT-4o actually follows the constraints set by the prompt. It is possible that GPT-4o continued to consider the chief complaint despite being told not to and would rely more on the patterns it learned during pretraining. Another possibility is that the combination of all other available variables could provide enough predictive power for KTAS classification, although this seems less likely given the strong importance of the chief complaints in the SHAP analysis applied on our RF model. These findings indicate that the adherence of GPT-4o to the prompts may not be absolute, which highlights a potential limitation when we rely on it for triage decision making.

Additionally, it turns out that some unconventional prompting methods can actually outperform zero-shot prompting. For example, prompts that involve a time limit or a math puzzle have shown to enhance performance. This suggests that adding an external task or prompting GPT-4o to engage in a more structured way of thinking before it answers, seems to improve its ability to reason more like experts' reasoning in KTAS classification.

However, the "no prior knowledge" prompt led to a better Weighted Kappa score compared to zero-shot prompting, which surprised us. We expected that asking GPT-4o to pretend that it had no prior knowledge of triage would result in worse performance. The fact that this did not happen suggests that GPT-4o might lean more on its existing medical knowledge rather than the specific context given in the prompt. This brings up a crucial question: how well does GPT-4o really adjust to the constraints in the prompts, and how much does it depend on its pre-trained knowledge even when told not to? If the model keeps relying on its internal knowledge despite clear instructions, it raises concerns about its

controllability, which could be an issue in high-stakes environments such as triage. Additionally, this could lead to biases in its decision making, as it could favor general medical trends over the unique details of each case, ultimately affecting predictability in clinical situations.

Even more unexpectedly, when GPT-4o was prompted with incorrect information in the adversarial prompt, it still managed to perform similarly to zero-shot prompting. In fact, its Weighted Kappa was even slightly better. This is fascinating because it suggests that the model can grasp the underlying trends in the data, even when it is given wrong information. Since GPT-4o's performance did not completely fail when providing it this prompt indicates that the model is again relying more on its internal knowledge rather than the information provided in the prompt. Such robustness may be a good quality for maintaining steady performance under differing triage conditions or when there are ambiguous cases present. However, this raises the same concern that we had with the "no prior knowledge" prompt: how much control do we truly have over GPT-4o's reasoning process? If the model continues to rely on its internal knowledge and learned patterns rather than strictly adhering to the prompt, it questions the extent to which we can guide its decision making.

Furthermore, all experimental prompts—with the exception of the complaint constraint prompts and the no prior knowledge prompt, which performed slightly better on KTAS level 4—achieved their highest F1 scores for KTAS level 1. This suggests that even unconventional or cognitively disruptive prompts can help GPT-4o recognize clear high-acuity patterns, likely due to strong associations learned from the data on which it is trained. KTAS level 5, however, consistently yielded the lowest F1 scores across all experimental prompts.

Self-Extracted Rule Prompting

When prompted with its own extracted triage rules, GPT-4o showed a noticeable drop in performance. This highlights an important aspect of how LLMs operate: while the model is great at making probabilistic inferences, it struggles to turn its internal decision making into clear, general rules. Essentially, the strength of the model lies in its ability to synthesize information through learned patterns rather than through transparent, rule-based logic. However, since GPT-4o may not be following its own triage rules for its predictions, there are reasons to worry about transparency in clinical decision support. If GPT-4o gives only explanations that are proximally consistent with human intuition rather than its true internal reasoning process, then its dependability as a decision support tool should be questioned. Further discussion on transparency concerns will be provided in Section 5.3.3.

At the class level, GPT-4o achieved moderately strong F1 scores for KTAS 1, KTAS 3, and KTAS 4, but its performance dropped for KTAS 2 and KTAS 5. While we see similar trends for all prompting techniques, the F1 score for KTAS 2 is significantly lower than every other prompting technique. This suggests that GPT-4o's self-extracted rules may fail to capture the specific clinical reasoning needed to identify moderately urgent cases. KTAS 2 cases may require careful integration of multiple subtle features that the extracted rules from the model do not adequately account for.

In summary, our findings show that SHAP-informed prompting significantly improved GPT-4o's performance, especially when combined with flexible rule exceptions, implicit-CoT reasoning, and fluent input. This suggests that GPT-4o may benefit from providing domain knowledge in its prompt. However, surprisingly, the experimental prompting techniques showed concerns about transparency, trustworthiness, and controllability. Furthermore, the performance drop when providing GPT-4o with its self-extracted rules in its prompt reveals a gap between GPT-4o's stated reasoning and its actual predictions. Additionally, both GPT-4o and the classical ML models struggled with KTAS level 5, likely due to feature overlap and ambiguity in low-acuity cases.

5.2 Comparison with Previous Research

This section reviews the applications of DT, RF, and XGBoost in previous research, followed by an overview of prior studies on prompt engineering techniques in triage. For both topics, we will compare existing findings with our study, highlighting similarities and differences.

5.2.1 Applications of DT, RF, and XGBoost in Triage

Accurate triage is crucial for optimizing ED operations. However, triage accuracy is influenced by multiple factors, such as ED overcrowding, the experience of emergency nurses, patient characterists, and work environment [6, 26, 141, 143]. Furthermore, research shows that the performances of established triage systems varies considerably [176] and that, despite rigorous training programs, reliance on human intuition can still lead to misclassification, demonstrating the limitations of current triage methodologies [54]. To address these challenges, ML models, including DT, RF, and XGBoost, have been explored to support and enhance ED triage decision making.

Several prior studies have demonstrated the effectiveness of RF and XGBoost in predicting triage outcomes. For instance, Yun et al. [173] implemented XGBoost to forecast critical care outcomes for patients in the ED, achieving an AUC of 0.861. Similarly, Jiang et al. [61] evaluated ML-based triage for patients suspected of having cardiovascular disease and found that XGBoost, RF, and gradient-boosted decision trees (GBDT) to perform well, with XGBoost achieving the highest AUC of 0.937.

Elhaj et al. [34] conducted a comparative study of nine ML models to predict ED triage, in which RF and XGBoost emerged as the best performers, with RF achieving 89% accuracy, recall, and F1-score. Likewise, Khanna et al. [68] employed different classifiers such as RF and XGBoost for COVID-19 severity prediction. Their RF model received the best recall of 83%, and used explainable AI methods SHAP and LIME to identify contributors such as respiratory rate and blood pressure. Furthermore, Aljameel et al. [5] applied RF and XGBoost to predict patient outcomes, with RF leading achieving 95% accuracy and an AUC of 0.99.

While earlier research has certainly made great strides in the application of ML models for triage tasks, most of it has focused on predicting critical care outcomes or hospital admissions rather than directly classifying triage levels [9, 57, 123]. That said, some studies have specifically looked into the use of ML for predicting triage levels. For instance, Jiang et al. [61] showed that ML models can help in triage decision making for patients suspected of having cardiovascular issues, relying solely on the data available at the time of triage. In their findings, the RF and XGBoost models achieved the highest AUC of 0.922. In a related study, Choi et al. [21] focused on training and comparing ML models for KTAS classification. They pointed out the practicality of using ML for direct triage level predictions, as outlined in the KTAS guidelines. Their results showcased strong discriminative ability, with XGBoost achieving an AUC of 0.937.

Our research takes a step further by not just predicting KTAS levels but also examining how well DT, RF, and XGBoost classify KTAS levels assigned by both registered nurses and expert clinicians.

When compared with previous studies that obtained AUCs over 0.90, our models recorded slightly lower AUC values (0.8941 for RF and 0.8818 for XGBoost on expert KTAS). This difference could stem from the unique aspects of our KTAS dataset. We worked with a relatively small sample of 1,264 patients, training our models on a subset of just 948 patients. This limited sample size probably affected how well the models could generalize to unseen patient data, leading to a lower performance compared to previous research. Moreover, even though we tried to address class imbalance using the ADASYN technique, the underlying imbalance in the dataset might have also played a role in the models' performance.

5.2.2 Applications of Prompt Engineering in Triage

Multiple studies have investigated how various prompting techniques can improve the way LLMs handle clinical data and assist in decision making under uncertainty.

For instance, Grabb [45] highlights the need for rigorous prompt engineering to ensure safe and effective LLM usage in mental health contexts, using an example in psychiatry. Another example is DeID-GPT, which illustrates how GPT-4 can be utilized for zero-shot medical text de-identification. This approach excels at protecting privacy by pinpointing and removing any identifying information, making it especially relevant for sharing healthcare data [88]. Similarly, HealthPrompt has made impressive strides by utilizing zero-shot learning for clinical NLP tasks. They have harnessed prompt-based techniques to effectively classify clinical texts without the need for any training data, highlighting how adaptable prompt learning can be in healthcare settings [139].

In the world of medical imaging, Milecki et al. [97] has launched a framework, MEDIMP, that utilizes LLMs to craft textual prompts designed for analysing renal transplant imaging data. Additionally, Wang et al. [156] employed prompt-based techniques to navigate glioma diagnosis using non-enhanced data, showcasing how prompts can adapt to overcome the hurdles of data collection limitations. Furthermore, by prompting ChatGPT-4.0 and Google GEMINI with detailed patient demographics, injury characteristics, and medical histories, Frosolini et al. [40] as-

sessed their triage decisions against those made by a tertiary referral center. This study revealed the potential of LLMs in maxillofacial trauma cases, as these models demonstrated moderate agreement with expert recommendations, accurately identifying injury severity and suggesting appropriate treatment steps.

Furthermore, Prompt-Eng explored the use of paired context-aware prompts in predicting medical outcomes, emphasizing prompt design for improved healthcare data interpretation [3]. Also, Han and Choi [50] created a multi-agent Clinical Decision Support System (CDSS) for emergency care using Llama-3-70b. By simulating roles such as Triage Nurse and Emergency Physician, their system not only performed accurate triage assessments based on KTAS but also demonstrated robust performance in areas like treatment planning and resource allocation. Lastly, a study by Williams et al. [163] took a closer look at how GPT-4 performs when it comes to assessing clinical acuity in an ED. They prompted the model to compare the clinical histories of patients, using records that included Emergency Severity Index (ESI) levels. In contrast to our task, where GPT-4o had to classify based on a 5-level classification task, they randomly selected pair and let their LLM predict which had a higher acuity level. Impressively, GPT-4 achieved an accuracy rate of 89% in predicting higher-acuity cases, which was close to the accuracy of physician reviewers. This suggests it could serve as a reliable tool for triage. However, it is difficult to compare their performances to ours, as we are dealing with a 5-level classification task instead of a 2-level classification task.

Work that is closely related to ours is by Nazary et al. [105]. They investigated how OpenAI's ChatGPT-3.5-Turbo can be used for clinical decision support. They introduced contextual prompts consisting of structured domain knowledge from applying SHAP on their best performing ML model, XGBoost. Their results show that when ChatGPT is given prompts that include this knowledge, it can perform similarly to traditional ML models, especially in zero-shot and few-shot contexts. Our study is in line with theirs, as we also investigated SHAP-informed insights into our prompt design. However, our SHAP-informed prompts did not outperform our classical ML models.

A clear distinction between their study and ours is that Nazary et al. [105] primarily concentrated on few-shot prompting techniques, testing various amounts of examples—0, 2, 4, 8, and 16. Their results showed that increasing the number of examples led to better performance, with their most effective prompt reaching an accuracy of 84.24%. In comparison, we only utilized two examples in our few-shot prompting and did not delve into how adding more might change things. Instead, our study concentrated on a broader array of prompt engineering techniques that go beyond just few-shot learning. Furthermore, it is worth mentioning that all of Nazary et al. [105]'s prompting methods reached accuracies exceeding 60%. Their best performing prompt even reached an accuracy of 84.24%. In contrast, our best performing prompt only achieved 60.44%. This difference can be attributed to the fact that we are dealing with a five-class classification task, whereas Nazary et al. [105] concentrated on binary classification tasks. In their work, they utilized ChatGPT-3.5-Turbo to evaluate heart disease risk based on various input attributes, categorizing individuals as either high-risk (with $\geq$50% artery narrowing, marked as '1') or low-risk (with $<$50% artery narrowing, marked as '0').

5.3 Ethical Implications of Applying GPT-4o in Triage

Effective triage is important to ensure that patients receive timely care, while also efficiently allocating medical resources. However, the quality of triage decisions often varies. This is due to factors such as clinician experience, fatigue, and the inherent complexity of the process [21, 35, 175]. Therefore, the increasing capabilities of LLMs introduce new promises in supporting healthcare providers. However, there are some ethical concerns that must be carefully considered. Four primary ethical concerns are discussed in this section: Accuracy and Reliability, Bias and Fairness, Transparency and Explainability, and Data Privacy and Security.

5.3.1 Accuracy and Reliability

Ensuring accurate and reliable LLMs like GPT-4o in triage systems is of critical importance since incorrect classification lead to misprioritization of patients. One significant challenge is that LLMs have a tendency to create hallucinations, wherein they generate information that is plausible but not true [106]. This is particularly problematic in triage, where accurate decisions must be made and based on evidence not to do any harm to patients. Furthermore, LLMs lack the individual patient's case context knowledge and are thus less trustworthy when dealing with complex or confusing symptoms [148]. In addition, if the training data becomes outdated, LLMs can drift, which can gradually diminish their performance and lead to inaccurate triage predictions [124].

Our findings highlight the current limitations of LLM-based triage predictions, since the best performing prompting technique from our research, SHAP + Exceptions + Implicit-CoT + Fluent Style prompting, achieved a Weighted Kappa of only 0.6186. This indicates moderate agreement with expert triage KTAS classifications, but a significant proportion of patients would still be misprioritized, making it unsuitable for real-world triage. Notably, this performance is even lower than our worst performing ML model, namely DT, which achieved a Weighted Kappa of 0.6328. Given that a Weighted Kappa of 1 indicates perfect agreement and a value near 0 suggests predictions are no better than chance, all ML models as well as GPT-4o result in suboptimal performance for deployment in high-stakes triage systems.

However, since our SHAP-informed prompts were based on the best performing RF model, they inherently reflect the limitations of that model, which achieved only 69.94% accuracy. As nearly one-third of predictions were incorrect, the SHAP-derived feature importances do not fully represent the true nuances in our KTAS dataset. This raises a key question: how would GPT-4o perform if guided by SHAP insights from an ML model with perfect accuracy? While our results suggest that neither LLMs nor ML models are accurate enough to be used in real-world triage, future research could explore whether an optimally trained ML model could enhance LLM-based predictions.

5.3.2 Bias and Fairness

The vast and varied textual data on which LLMs are trained inevitably contain societal biases, inconsistencies, and potential misinformation [125]. Because of this, these models might unintentionally absorb and reinforce the biases that already exist in their training data. This concern is especially critical in medical applications, such as triage, where patients could be misprioritized due to biased predictions. If an LLM mistakenly favors certain groups disproportionately due to biases in its training data, it could cause delays in urgent care or inappropriate allocation of resources. This is supported by Pan et al. [116], who found that LLM-generated misinformation can cause actual world harm. Therefore, it is important to ensure that classifications made by LLMs are fair to prevent harmful outcomes for patients.

An important uncertainty in using GPT-4o for KTAS classification is that OpenAI did not provide details regarding its training process [25]. However, if we do not know the specific data on which it was trained, it is difficult to assess whether it has been trained on sufficient high-quality triage information. If training datasets lack proper triage examples, which may be the case, then the model will resort to general medical experience, which although helpful is not at all a part of the formally structured decision process that takes place in triage. Even if trained using triage guidelines, one cannot guarantee they are as compliant as KTAS guidelines. Since the KTAS system has an organized paradigm unique to emergency care in South Korea, a discrepancy between learned knowledge from the model and the actual triage criteria could lead to incorrect predictions. Therefore, rigorous testing and removal of bias methods are needed to ensure that the model makes clinically applicable triage decisions.

OpenAI has taken steps to make its LLMs more fair [142]. OpenAI, for example, employs filtering techniques to remove toxic material from the training set [111] and they also introduced WebGPT [103] in the alignment process, a framework to assist human reviewers in identifying and correct errors in LLM output. However, since AI model biases will inevitably originate from the initial data [140], the value of these techniques to avoid bias in triage decision making is unclear.

A further consideration is that Western-centric data are the basis for training LLMs like GPT-4o [107, 172]. This can limit their use in non-Western clinical environments such as the prediction of triage acuity by the use of KTAS guidelines. Thus, an interesting direction for future research may be in the use of other models like DeepSeek. Since DeepSeek was trained with a Chinese worldview [66], its clinical decision making could potentially be different from GPT-4o. This suggests that DeepSeek could be more appropriately suited for KTAS classification, since its training data could more effectively reflect Asian healthcare systems' priorities and protocols. Investigating region-specific models such as these could provide valuable insight into the effect of different training data distributions on triage decisions and fairness, but this exploration lies beyond the scope of our current study.

Mitigating biases requires careful curation of training data and the use of bias-mitigation methods to reduce discriminatory predictions. Furthermore, an effort

combining domain experts, data scientists, and ethicists should be taken to create guidelines for the identification and rectification of bias in LLM-based healthcare applications [106].

5.3.3 Transparency and Explainability

Maybe the most important challenge for the deployment of an LLM such as GPT-4o in a triage system is their inherent lack of transparency and explainability. Unlike classical ML models that will at least be subject to some degree of interpretability through feature importance rankings such as SHAP, LLMs are "black-box" models whose internal reasoning processes are not transparent [48]. This lack of transparency is worrying as trustworthiness is an essential demand for high-stakes applications such as triage [10]. Without comprehensible explanations of why an LLM places a particular patient in a specific triage category, clinicians cannot confirm or sanction its suggestions [152].

Explainability of AI is important for several reasons, such as assisting users to identify patterns, explaining the reasons for incorrect predictions, promoting user trust in model predictions, allowing bias detection, guaranteeing compliance with the law, and acting as a precaution against overfitting [2, 46, 104]. Thus, not being able to fully comprehend the rationale for LLM predictions continues to be a basic hindrance to clinical adoption.

Ideally, an explainable triage model would tell end-users exactly how a particular patient was assigned to a particular KTAS level. However, our research suggests that GPT-4o cannot provide understandable explanations for why it took a specific decision. Interestingly, when we extracted triage rules from GPT-4o and fed them back into the model for classification, performance dropped significantly compared to zero-shot (Weighted Kappa: 0.2841 vs. 0.4374). This suggests that GPT-4o does not work consistently by explicit rule-based thinking but by the internal learned weightings during training data that are not explainable. Remarkably, even if we flipped KTAS levels in the prompt, the model still performed better than zero-shot (0.4565 vs. 0.4374). This leads us the question: what type of decision rules does GPT-4o use, if any? If the model actually relied on structured rules, its performance on using its own derived rules would be at least as good as its zero-shot performance. The fact that this was not the case, suggests that GPT-4o might not use explicit rule-based reasoning or that its own decision making is based on deeper, impenetrable patterns we cannot reverse-engineer easily. Knowing the model's true decision process is still an open question and a primary area for future research.

In addition, under an instruction to use chief complaint only for KTAS classification, performance increased significantly above zero-shot (Weighted Kappa: 0.5352 vs. 0.4374). This suggests that the model was making better decisions when it was told to constrain its predictions to this variable. But when we reversed this approach and asked GPT-4o to use all variables except the chief complaint, performance improved as well (Weighted Kappa: 0.4704 vs. 0.4374). This seems counterintuitive, as, if chief complaints were indeed one of the most influential variables for triage classification, we would expect that removing them would decrease

performance. This indicates that GPT-4o might not be systematically weighing variables in a way that is useful for clinical settings and leads to a vital question: how can we trust an AI model if it is not able to explain the decision making process that let to its predictions?

Despite these challenges, one effective strategy was the use of SHAP-informed prompts. By including SHAP values from our best performing RF model, we were able to significantly improve GPT-4o's performance. This indicates that LLMs can be nudged towards better predictions if important features were stated explicitly. However, this result should be tempered with caution because of the other inconsistencies that were observed in our experiments. Furthermore, since our best performing ML model was only 69.94% accurate, we must question whether higher performance could be achieved even if GPT-4o had been trained on SHAP values from a perfectly predicting model. This scenario was not feasible in our study.

Although these results suggest that we can, to some extent, shape GPT-4o's decision making, the larger issue is still with them: LLMs struggle to provide a structured, open explanation of reasoning for their triage predictions. Future research should aim to further optimize prompt strategies, determine if LLMs truly internalize structured knowledge, and analyse whether enhanced SHAP-driven methods yield a more interpretable AI-augmented triage system.

5.3.4 Data Privacy and Security

As LLMs like GPT-4o continue to be more integrated in healthcare applications, data privacy and safety issues are increasing. Especially in healthcare, patient information has to be protected. If a model cannot be made secure, this could result in breaches of clinical confidential data, which could have catastrophic consequences [98, 142].

One concern is that LLMs, such as GPT-4o, may inadvertently collect, store, or disclose confidential health information. This is a threat in terms of unauthorized use, breaches, and accidental storage [102]. Although not directly trained on patient clinical data, LLMs could still produce inaccurate or privacy-violating outputs [55]. On top of that, LLMs can be misled to perform harmful tasks [69]. Additionally, deidentified information remains vulnerable to reidentification using cross-matching with external sources [33]. These issues bring up serious concerns about compliance with privacy regulations like HIPAA [109, 153].

To address these problems, OpenAI created a Red Teaming Network with the goal to identify security weakness before being deployed [112]. However, LLMs remain susceptible to adversarial manipulations such as prompt injection attacks [162], which can lead to unwanted behavior and patient harm. Furthermore, the way roles are assigned and system parameters are configured plays a crucial role in shaping the behavior of LLMs. The misuse of them can lead to bias or the creation of harmful content [31].

The last thing we want to point out is that legal and ethical limitations have prevented health care units from sharing identifiable patient information with freely available LLMs. Although these restrictions are essential for protecting patient privacy, they also limit the ability of the model to generalize effectively to real-

world clinical scenarios [153]. Without access to real patient data, models may lack the nuanced understanding necessary for accurate decision support. This tension between privacy protection and model optimization remains a major barrier to the effective deployment of LLMs in healthcare settings.

The uncertainties when using LLMs such as GPT-4o in triage systems reveal several ethical concerns. Although LLMs are capable of producing intricate patterns of reasoning, their lack of transparency raises critical questions regarding their dependability in high-risk situations such as triage. Despite the use of approaches such as SHAP-informed prompting, the underlying reasoning processes in GPT-4o do not become transparent and, therefore, are not suitable for use in triage. These difficulties underscore the need for an AI system that integrates the capabilities of LLMs with a structured rule-based system to ensure consistency and explainability for triage outcomes. One possible solution is the incorporation of BOID, which reflects how clinicians make decisions, in LLMs.

5.4 Incorporating BOID into LLMs

By integrating the structured rule-based default reasoning of the BOID architecture into an LLM, we leverage LLM capabilities while enabling the traceability of which default rules take precedence when conflicting beliefs, obligations, intentions, and desires arise during triage decision making. This integration will ensure explainability of decisions such that healthcare providers can comprehend the rationale behind the recommendation provided by the system. Through a clear justification of every decision, a BOID-LLM framework could facilitates trust in such a way that healthcare providers can critically consider the recommendation. Because of this, healthcare providers can also maintain a continuous quality control process, making provision for appropriate modifications to be made in the system so that the system remains in sync both with altered clinical guidelines and social contexts.

Furthermore, a BOID-LLM system could enhance productivity by reducing the burden of routine decision making, which allows healthcare providers to focus on other critical aspects of care, such as direct treatment and effective patient communication. In addition, with a BOID-LLM model, there is greater consistency in recommendations. Under strict guidelines to follow, the hybrid model can reduce the risk of human errors due to inexperience, fatigue, and the nature of complexity inherent in triage decisions. It can also help limit the influence of personal bias in high-emotional situations—such as when a triage clinician is asked to assess a close friend or relative, as we illustrated earlier in our example—by having decisions made based upon prioritized responsibilities and traceable rules rather than subjective opinion. Consistent applying these principles is especially important in triage situations, where the decisions have immediate consequences for patient care.

Moreover, a BOID-LLM as decision support tool could streamline triage processes, prevent unnecessary tests or hospital admissions, and provide faster triag-

ing, resulting in minimizing treatment delays, lowering the risk of costly medical errors, and ensuring optimal use of available resources.

In conclusion, the a BOID-LLM system has the potential to be a big step in the field of healthcare, by not only enhancing decision making efficiency and trustworthiness of the model, but also by improving the general quality of care for patients.

5.5 Limitations and Future Research

The original vision for this research was to create this BOID-LLM framework, which ultimately functions as a decision support tool for healthcare providers by integrating the structured reasoning of BOID with the expansive data-handling capabilities of LLMs. The goal was to develop a system capable of automated decision making in triage, with decisions that were both explainable and traceable.

One possible path toward realizing this would have been to extract reasoning rules corresponding to each of the BOID components from an LLM like GPT-4o using carefully designed prompts. In theory, this could allow the construction of a synthetic dataset representing BOID. However, this approach would be unreliable due to the black-box nature of LLMs: while the model can generate plausible-sounding outputs, there is no guarantee that these outputs reflect valid clinical reasoning. Because of this, we decided not to pursue this synthetic rule-extraction route.

Instead, we found an interesting open-source dataset containing dual KTAS labels, one assigned by registered emergency nurses and the other by triage experts [182]. This structure allowed us to explore not only the general problem of KTAS classification, but it also enabled us to compare how well our classical ML models could replicate triage decision making patterns of nurses and experts, respectively.

Since this dataset includes only final KTAS classifications and not the underlying reasoning used to arrive at those classifications, we adopted the simplifying assumption that these decisions were primarily obligation-driven and we investigated whether GPT-4o could simulate this reasoning process using various prompt engineering techniques.

Despite this refined scope, several methodological limitations must be acknowledged. First, ADASYN was employed in this research to address class imbalance, particularly for minority KTAS 1 and KTAS 5 cases. While this approach enhances classification performance for minority classes and avoids overfitting risk, it can introduce noise when the synthetic samples fail to accurately reflect the actual data distribution [114]. This might have affected model performance in a way that is not entirely typical of actual South Korean ED distributions. Future work could explore if other resampling strategies yield more robust results.

Also, handling missing values is another limitation. Mode imputation was used for missing numerical attributes. This is a straightforward method that works well for categorical variables and keeps the distribution intact, but it can introduce bias if the most frequent category is overrepresented [108]. Exploring more advanced imputation techniques, such as K-Nearest Neighbors (KNN) Imputation or Multi-

ple Imputation by Chained Equations (MICE), could be beneficial to see how they impact classification performance [108].

A further possible limitation is the inclusion of the "group" variable, indicating whether a patient visited the Local ED or Regional ED. This variable might insert biases into model predictions, as differences between the two facilities in triage processes, available resources, and patient populations could impact KTAS assignments. Our RF model's SHAP analysis revealed that "group" was a predictor of model predictions, suggesting the models may be picking up on systemic differences between the two hospitals rather than purely medical factors. This raises concern about the generalizability of our models, to the extent their predictions are conditional on institutional trends rather than patient-level urgency.

Additionally, it would be interesting to investigate alternative ways of generating structured rules from the dataset, rather than just relying on SHAP-based feature importance. One potential method is Probabilistic Inductive Logic Programming (PILP) [29], which allows for the extraction of probabilistic logical rules from data. This technique could be used to develop probabilistic logic rules from the dataset, which could then be integrated into prompts for GPT-4o, much like the SHAP-informed prompting used in this study. Diving into PILP was outside the scope of this research, but it opens up a path for future exploration.

When it comes to LLMs, a notable limitation when using GPT-4o in predicting KTAS levels could be that GPT-4o is trained on Western-centric data [107, 172]. This may limit their use in non-Western clinical environments such as the prediction of triage acuity for South Korean patients. Therefore, it would be very interesting to see how other LLMs would perform on our KTAS classification task. In particular DeepSeek R1, since this LLM was trained with a Chinese worldview [66], suggesting that it could better reflect Asian healthcare systems' priorities and protocols. It would be interesting to quantify how well DeepSeek R1 performs for KTAS classification. By examining its decision making patterns, biases, and clinical relevance in a KTAS context, we could gain valuable insights and alternative viewpoints on LLM-assisted triage. Unfortunately, during this research, we could not access a DeepSeek API key, which limited our investigation. However, now that the API key is available, future studies could delve into how DeepSeek would perform when predicting KTAS levels.

Moreover, while we assessed GPT-4o's capabilities in triage classification using prompt engineering, we did not explore the potential benefits of fine-tuning the model with KTAS data. Given that GPT-4o was probably not specifically trained on KTAS guidelines, it is reasonable to think that a fine-tuned model would yield better results than one that depends solely on prompt engineering. Unfortunately, fine-tuning was outside the scope of this study due to constraints in time and resources. Future studies might want to look how effective fine-tuning GPT-4o could be in predicting KTAS levels.

Furthermore, although we did compare the rules extracted from GPT-4o with SHAP-based feature importance from ML models, we did not look at whether real-world triage experts would actually agree with or use these rules in their day-to-day work. The inclusion of South Korean emergency physicians and triage nurses in a validation study could help determine whether the rules generated by

GPT-4o are clinically meaningful or whether they diverge from expert reasoning. This would also help to address concerns about implicit biases in LLM-generated decision rules and their applicability in high-stakes environments.

Although we did not use Retrieval-Augmented Generation (RAG) in this study, we recognize its potential as a promising direction for future work. A RAG-based setup could allow GPT-4o to ground its responses in external knowledge sources, such as real-world clinical guidelines or case databases. This could improve the reliability and consistency of its outputs and support more faithful rule generation aligned with actual triage protocols. However, in this study, we were specifically interested in how GPT-4o would behave without the support of retrieval mechanisms, relying solely on its internal knowledge and prompt structure.

But ultimately, the most promising path forward lies in developing a BOID-LLM, a system that integrates the structured, explainable default reasoning of the BOID with the advanced abilities of LLMs. This work has taken an early step in this direction and future research could build on this foundation to create a decision support tool for healthcare providers in triage and beyond that is transparent and clinically trustworthy while leveraging the strengths of LLMs.

6 Conclusion

In this study, we compared the performance of classical ML models against GPT-4o for KTAS classification, evaluating their performance under different conditions and prompt engineering techniques. In this concluding section, we will go through our four subquestions one-by-one, after which we will try to answer our main research question.

Among the ML models tested, RF performed slightly better than XGBoost and DT for predicting both registered nurse and expert KTAS levels. However, all models performed poorly for KTAS Level 5, likely due to class imbalance and complex feature interactions for this acuity level. Although RF resulted in the best performing model, its performance was still suboptimal. This shows that even our best classical ML model is far from perfect agreement with expert decisions.

Applying SHAP analysis to our RF model identified what feature values influenced the prediction of the different KTAS levels. In higher acuity, leading features were unresponsiveness, low oxygen saturation, cardiac arrest, hypotension, and arrival by ambulance, aligning with common medical knowledge. In lower acuity, non-urgent chief complaints, such as medication refill and follow-up, and stable vital signs dominated. Unexpectedly, SHAP results showed that female patients were positively associated with KTAS 1 and 3, while male patients were more associated with KTAS 2. Although these patterns may reflect genuine patterns in our dataset, they still warrant caution and further investigation into possible gender-related bias in triage decision making.

The extracted triage rules from GPT-4o partially aligned with the results from SHAP analysis. However, its simplified rules failed to capture complex feature interactions, particularly for NRS pain scores. The rules also omitted relevant variables, such as mode of arrival, age of patient, and presence of injury, which were significant according to SHAP analysis.

When looking at the prompting techniques, standard prompting techniques showed little to no improvement over zero-shot, contradicting with our hypothesis. This suggests that these techniques might just be adding redundant information that is not necessary in our task of predicting urgency levels according to the KTAS guidelines. However, as expected, SHAP-informed prompting enhanced GPT-4o's performance, particularly when combined with flexible rule exceptions, implicit-CoT reasoning, and fluent-style input. The best performing SHAP-informed prompt achieved a Weighted Kappa of 0.6186, which was the result of its improved performance for predicting KTAS levels 2 and 3, where other techniques underperformed. However, even the performance of this best performing prompting technique was still lower than the performance of all our classical

ML models. This indicates that GPT-4o was less reliable for KTAS classification than our classical ML models.

Experimental prompts produced mixed results. While emotional-based, reward-based, and nonsensical-based prompts had no effect, constraint-based prompts improved performance, while we expected the opposite to happen. Surprisingly, the prompt instructing GPT-4o to assume it has no prior triage knowledge as well as the adversarially misleading prompt did not degrade performance. These results suggest that GPT-4o relies more on learned medical knowledge than prompt instructions, raising concerns about controllability and the model's ability to follow explicit user constraints.

One of the primary problems was observed when GPT-4o was asked to classify using its own extracted rules, as its classification performance significantly dropped. This suggests that GPT-4o does not actually follow its own generated rules when making triage decisions, but rather uses implicit probabilistic reasoning. However, as GPT-4o may not be following its own triage rules for its predictions, this shows a gap between how the model claims to reason and how it actually reasons, which gives reasons to worry about its transparency. If GPT-4o gives only explanations that are proximally consistent with human intuition rather than its true internal reasoning process, then its predictions cannot be trusted as an interpretable decision support tool.

In short, while GPT-4o shows some capability in classifying KTAS levels, its performance does not reach the standard set by our classical ml models. SHAP-informed prompting offers some improvement, but still falls short of matching their effectiveness. Furthermore, GPT-4o showed inconsistencies in its extracted triage rules and showed a gap between how it claims to reason and what it actually predicts. In addition, GPT-4o seems to rely on general medical knowledge rather than explicit instructions in the prompt. These findings suggest that GPT-4o, when used without any external retrieval mechanisms and relying solely on prompt-based interaction and its internal knowledge, is not yet ethically or practically viable as a decision support tool in triage. Exploring hybrid AI frameworks, such as a BOID-LLM that integrates structured, rule-based default reasoning, represent a highly promising direction for future research, with the potential to enhance fairness, transparency, and clinical applicability in triage settings.

Appendices

A Computational Setup

Preprocessing and analysis were conducted using Python 3.13.2 in a Jupyter Notebook environment running on a local Windows 10 system. The computational setup included an AMD Ryzen 7 7730U processor with integrated Radeon Graphics. Jupyter Notebook was launched within a virtual environment. No dedicated GPU acceleration was used.

B Exploratory Data Analysis

B.1 Methodology of EDA

To examine relationships between variables, we used the following metrics [145]:

- **Numerical-Numerical Correlations:** Computed using Pearson correlation, which measures linear relationships between continuous variables.
- **Categorical-Categorical Correlations:** Assessed using Cramér's V with bias correction, calculated from confusion matrices to evaluate the strength of association between nominal variables.
- **Numerical-Categorical Correlations:** For binary categorical variables, Point-Biserial Correlation was used. For multi-class categorical variables, Eta-Squared was calculated by decomposing variance to determine the proportion explained by the categorical variable.

Using these ensured an accurate handling of mixed data types by dynamically selecting the appropriate correlation technique based on the variable types.

B.2 Results of EDA

The results of the EDA, as presented in Figure B.1, reveal relationships between the variables in the dataset. This analysis incorporated numerical, categorical, and mixed-type correlations using Pearson correlation, Cramér's V, and point-biserial or eta squared correlations, depending on the types of variables under consideration [145]. Associations with an absolute value greater than 0.20 are highlighted in bold.

Among numerical variables, systolic blood pressure (`SBP`) and diastolic blood pressure (`DBP`) displayed a strong positive correlation ($r = 0.78$), consistent with their physiological interdependence. Furthermore, pain levels, as measured by the `NRS_pain` score, were strongly associated with the presence of `Pain` ($r = 0.89$), which seems logical as `Pain` = 0 always comes together with `NRS_pain` = 0. Furthermore, age exhibited negative correlations with variables `registered nurse KTAS` (r = - 0.21) and `expert KTAS` (r = - 0.23), suggesting that it a higher age might result in lower, or more urgent, KTAS scores in this dataset. Other numerical features, such as age and various vital signs, demonstrated weaker correlations with each another, reflecting the independent contributions of these

features to the dataset. `Registered nurse KTAS` and `expert KTAS` themselves exhibited a high correlation ($r = 0.88$), indicating a strong agreement between nurses and experts in assigning urgency levels. This was also observed in Table 3.18.

The categorical variables, using Cramér's V, revealed moderate correlation between `complaint_category` and `Injury` ($V = 0.58$), `Mental state` ($V = 0.43$), `Pain` ($V = 0.56$), and `NRS_pain` ($V = 0.52$).

Mixed-type correlations, computed using point-biserial and eta-squared methods, further highlighted the interactions between numerical and categorical variables. The multiclass numerical category `registered nurse KTAS` and `expert KTAS` both revealed moderate associations with the category `complaint_category`, ($\eta^2 = 0.44$ and $\eta^2 = 0.46$, respectively). These findings suggest that the type of complaint reported by patients significantly influences the triage decisions made by both nurses and experts.

Table B.1: Lower Triangular Correlation Matrix with Variable Indices

		I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII
I	Group	**1.00**																	
II	Sex	0.00	**1.00**																
III	Age	0.14	- 0.05	**1.00**															
IV	Patients number per hour	- 0.09	0.03	0.02	**1.00**														
V	Arrival mode	0.11	0.01	0.07	0.01	**1.00**													
VI	Injury	- 0.03	0.08	**- 0.20**	0.03	0.00	**1.00**												
VII	Mental	0.00	0.01	0.01	0.00	0.16	0.00	**1.00**											
VIII	Pain	0.05	- 0.01	- 0.18	- 0.02	0.04	0.18	0.05	**1.00**										
IX	NRS_pain	- 0.01	0.01	- 0.16	- 0.03	0.02	0.18	0.05	**0.89**	**1.00**									
X	SBP	**- 0.29**	0.03	0.05	- 0.01	0.03	0.02	0.06	0.01	0.04	**1.00**								
XI	DBP	- 0.10	0.06	- 0.02	0.00	0.01	0.01	0.07	0.04	0.03	**0.78**	**1.00**							
XII	HR	- 0.09	0.06	- 0.09	- 0.02	0.00	- 0.02	0.01	- 0.07	- 0.05	0.03	0.05	**1.00**						
XIII	RR	**0.34**	- 0.03	0.10	0.06	0.05	- 0.05	0.05	- 0.06	- 0.07	- 0.12	- 0.04	0.13	**1.00**					
XIV	BT	**- 0.23**	0.00	- 0.06	0.02	0.00	- 0.09	0.02	- 0.04	- 0.02	0.11	0.04	**0.25**	0.03	**1.00**				
XV	Saturation	- 0.15	- 0.04	- 0.19	0.01	0.06	0.04	**0.27**	0.15	0.14	**0.30**	**0.26**	- 0.06	**- 0.22**	0.09	**1.00**			
XVI	Complaint category	0.13	0.06	0.16	0.03	0.19	**0.58**	**0.43**	**0.56**	**0.52**	0.06	0.06	0.10	0.14	**0.20**	**0.20**	**1.00**		
XVII	registered nurse KTAS	- 0.15	- 0.02	**- 0.21**	0.07	0.14	0.18	0.10	0.17	0.09	0.10	0.08	0.01	**- 0.20**	0.03	**0.31**	**0.44**	**1.00**	
XVIII	expert KTAS	- 0.19	0.00	**- 0.23**	0.06	0.13	**0.21**	0.13	0.19	0.08	0.13	0.11	0.01	**- 0.24**	0.01	0.35	**0.46**	**0.88**	**1.00**

C Prompts

C.1 Prompt Rule Extraction

```
import openai
import os
import pandas as pd
import json

# Set your API key
openai.api_key = os.getenv("OPENAI_API_KEY")

# Load sample data
df = pd.read_csv("path_to_dataframe")

# Define role-based system message
user_prompt = """
    You are tasked with formulating a highly detailed and accurate set of rules for determining patient urgency levels (1 = highest urgency, 5 = lowest urgency) based on the Korean Triage and Acuity Scale (KTAS).
    Your goal is to establish a comprehensive and hierarchial set of rules that:
    - maximizes prediction accuracy of KTAS levels.
    - distinguishes patients with high granularity using their clinical features.
    - covers all possible patient presentations, ensuring no overlap between KTAS levels.
    - optimizes the rules such that they match real-world triage decisions as closely as possible
    - consider various patient variables and identify exceptions to these rules.

    ### Patient Variables to use:
    - Group: Local ED or Regional ED
    - Sex: Male or Female
    - Age: 16-96
    - Patients number per hour: Number of patients arriving at the same hour as the patient
    - Arrival mode: Private Vehicle, Private Ambulance, Public Ambulance, Walking, or Other
    - Injury: Presence of injury - yes or no
    - Mental: Alert, Pain Response, Verbal Response, or Unresponsive
    - Pain_level: Presence of pain - yes or no
```

```
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = worst possible pain)
    - SBP: Systolic Blood Pressure (mmHg)
    - DBP: Diastolic Blood Pressure (mmHg)
    - HR: Heart Rate (beats per minute)
    - RR: Respiratory Rate (breaths per minute)
    - BT: Body Temperature (C)
    - Saturation: Oxygen saturation percentage
    - Complaint_category: Abdominal Pain, Back Pain, Chest Pain, Ear/Nose/Throat Problems, Traumatic Injuries Specific (FT), Convulsions; Seizures, Dizzy, Procedure, Respiratory Problems, Fainting/Syncope, Hemorrhage, Dental; Toothache, Weakness, Foreign Body, Fever, Unconsciousness, Abdominal Problems, Substance Abuse, Psychiatric/Social Problems, Body Aches, Follow-Up, Headache, Eye Problem, Burns, Peripheral Vascular/ Leg Pain, Neck Pain, Flank Pain, Diabetic Problems, Neurological Complaint, (Specific) diagnosis (FT), Laceration, Medication Refill, Skin Complaint/Trauma, Assault/Rape, Unknown Problem, Cardiac Arrest, Genito-Urinary Problem, Allergies/ Hives/Med Reaction/Sting, Orthopedic Injury, Obstetrical Problem, Fluid/Nutrition Alteration, Gynecological Problem, or Cardio-Vascular Complaint.

    ### Instructions:
    1. Generate detailed, mutually exclusive, and collectively exhaustive rules for each KTAS level (1 through 5)
    2. Prioritize rules from most critical to least critical
    3. Include multiple conditions per rule to make finer distinctions
    4. Account for exceptions where patients should be classified differently despite matching a general rule
    5. Ensure the rules are realistic and aligned with clinical best practices


    ### Output Format:
    Rules for KTAS Level 1: {}
    Rules for KTAS Level 2: {}
    Rules for KTAS Level 3: {}
    Rules for KTAS Level 4: {}
    Rules for KTAS Level 5: {}

    Provide only the list of rules without additional explanations or commentary. The rules should have the following structure: IF ... (=/!=/>/<) ... (AND/OR ... (==/>/<) ...)* - (1/2/3/4/5).
    Meaning of "*": combine multiple conditions using AND or OR, as required to define the appropriate urgency level.
"""

system_prompt = """You are an AI expert in emergency medicine."""

response = openai.chat.completions.create(
    model="gpt-4o",
    messages=[
```

```
        {"role": "system", "content": system_prompt},
        {"role": "user", "content": user_prompt}
    ],
    temperature=0
)

response_text = response.choices[0].message.content

print(response_text)
```

Listing C.1: Prompts Rule Extraction

C.2 General Workflow all Prompting Techniques

```
# System prompt assigned for each prompting technique.
system_prompt = """ """ # Specific prompt for each prompting technique specified in Appendix C.3

# Function to extract and format patient data
def patient_description(df_batch):
    """
    Converts a DataFrame batch into a structured list for LLM processing.
    """

    patients_list = []

    for _, row in df_batch.iterrows():
        patient_data = {
            "Group": row["Group"],
            "Sex": row["Sex"],
            "Age": row["Age"],
            "Patients_per_hour": row["Patients number per hour"],
            "Arrival_mode": row["Arrival mode"],
            "Injury": row["Injury"],
            "Mental_status": row["Mental"],
            "Pain_level": row["Pain"],
            "NRS_pain": row["NRS_pain"],
            "SBP": row["SBP"],
            "DBP": row["DBP"],
            "HR": row["HR"],
            "RR": row["RR"],
            "BT": row["BT"],
            "Saturation": row["Saturation"],
            "Complaint_category": row["complaint_category"],
        }
        patients_list.append(patient_data)

    return patients_list

# Initialize result storage
all_predictions = []
```

```
# Process 10 patients at a time
BATCH_SIZE = 10

# Iterate through dataset in batches
for i in range(0, len(df), BATCH_SIZE):
    # Select a batch
    df_batch = df.iloc[i:i + BATCH_SIZE]

    # Convert to structured format
    patients_batch = patient_description(df_batch)

    # Define the user prompt with patient batch
    ### Note: The user_prompt is different for fluent style prompting, see Appendix C.7 ###
    user_prompt = f"""
    Patients:
    {json.dumps(patients_batch, indent=2)}
    """

    # Call the GPT-4o API with the selected system prompt
    response = openai.chat.completions.create(
        model="gpt-4o",
        messages=[
            {"role": "system", "content": system_prompt},
            {"role": "user", "content": user_prompt}
        ],
        temperature=0
    )

    # Extract and print response
    response_text = response.choices[0].message.content
    print(response_text)
```

Listing C.2: Batch processing and API call for GPT-4o

C.3 Complete GPT-4o Prompts

C.3.1 Standard Prompting

Zero-shot Prompting

```
system_prompt = """
    You will be provided with the values of 10 patients, each with 16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = maximal pain).
```

```
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.3: Zero-shot Prompting

Few-shot Prompting

```
system_prompt = """
You will be provided with the values of 10 patients, each with 16 variables.
Predict the urgency level (1 = highest urgency, 5 = lowest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS).

Some abbreviations in the data:
-NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = maximal pain).
-SBP: Systolic Blood Pressure.
-DBP: Diastolic Blood Pressure.
-HR: Heart Rate.
-RR: Respiratory Rate.
-BT: Body Temperature.

### Output Format:
Return ONLY a valid JSON list containing objects with the field:
-"KTAS level": (int) A single integer between 1 and 5.

### Example 1:
Patient:
{
"Group": Regional ED,
"Sex": male,
"Age": 68,
"Patients number per hour": 10,
"Arrival mode": Private Vehicle,
"Injury": no,
"Mental State": Alert,
"Pain": yes,
"NRS_pain": 6,
"SBP": 160.0,
"DBP": 80.0,
"HR": 66.0,
"RR": 20.0,
"BT": 36.2,
"Saturation": 99.0,
```

```
"complaint_category": Flank Pain
}
Prediction:
{
"KTAS level": 3
}

### Example 2:
Patient:
{
"Group": Regional ED,
"Sex": female,
"Age": 20,
"Patients number per hour": 4,
"Arrival mode": Private Vehicle,
"Injury": no,
"Mental State": Alert,
"Pain": no,
"NRS_pain": 0,
"SBP": 120.0,
"DBP": 70.0,
"HR": 78.0,
"RR": 24.0,
"BT": 36.0,
"Saturation": 98.0,
"complaint_category": Neurological Complaint
}
Prediction:
{
"KTAS level": 4
}

### Now classify the following patients.
"""
```

Listing C.4: Few-shot Prompting

Role Prompting

```
system_prompt = """
    You are the head emergency department triage nurse with 20+ years of experience using the Korean Triage and Acuity Scale (KTAS).
    You will be provided with the values of 10 patients, each with 16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = maximal pain).
    - SBP: Systolic Blood Pressure.
```

```
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.5: Role Prompting

Implicit Chain-of-Thought

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
    16 variables.
    Your task is to analyze each patient's medical condition step-
    by-step.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.6: Implicit Chain-of-Thought

Fluent Style Prompt

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
    16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
```

```
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""

user_prompt = "A {row["Age"]}-year-old {row["Sex"]} patient from {
    row["Group"]} group arrived at the emergency department via {
    arrival_mode} during a time when {row["Patients number per hour
    "]} patients per hour were being admitted. The chief complaint
    of the patient is {chief_complaint}, and currently exhibits a
    mental status of {row["Mental"]}. The patient {("reports a pain
     level of " + str(row["NRS_pain"]) + " out of 10" if row["Pain"
    ] == "yes" else "denies pain")}. Additionally, the patient {("
    has sustained an injury" if row["Injury"] == "yes" else "has no
     reported injury")}. Vital signs include a systolic blood
    pressure of {row["SBP"]} mmHg and a diastolic blood pressure of
     {row["DBP"]} mmHg, with a heart rate of {row["HR"]} bpm, a
    respiratory rate of {row["RR"]} breaths per minute, a body
    temperature of {row["BT"]}C, and an oxygen saturation of {row["
    Saturation"]}%."
```

Listing C.7: Fluent Style Prompt

C.3.2 SHAP-Informed Prompting

Prompting with SHAP Analysis

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
     16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
     maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    SHAP analysis identified key variables influencing KTAS levels
    . The order of variables reflects their importance, and the
    specific values listed are those that increase the probability
    of that KTAS level.
```

```
KTAS 1:
- Mental_Unresponsive = True
- Saturation = low
- SBP and DBP = low
- complaint_category_Psychiatric/Social Problems = True
- Arrival mode_Private Ambulance = True
- Pain = no
- Mental_Alert = False
- Arrival mode_Public Ambulance = True
- Mental_Pain Response = True
- Age = high
- Arrival mode_Private Vehicle = False
- NRS_pain = low
- HR = high
- complaint_category_Cardiac Arrest = high
- Sex = female
- complaint_category_Respiratory Problems = True
- Patients number per hour = low
- RR = high
- Injury = no

KTAS 2:
- complaint_category_Chest Pain = True
- Arrival mode_Private Vehicle = False
- Group = Regional ED
- Arrival mode_Private Ambulance = True
- Mental_Unresponsive = False
- NRS_pain = low or high
- Age = high
- HR = low
- complaint_category_Abdominal Pain = False
- Mental_Alert = False
- Saturation = average
- Sex = female
- RR = high
- complaint_category_Psychitric/Social Problems = True
- complaint_category_Weakness = High
- Mental_Pain Response = True
- complaint_category_Hemorrhage = True

KTAS 3:
- NRS_pain = low or high
- Injury = no
- Pain = no
- complaint_category_Abdominal Pain = True
- SBP = high
- Mental_Unresponsive = False
- complaint_category_Dizzy = True
- Group = Regional ED
- Saturation = high
- Age = high
- DBP = high
- complaint_category_Chest Pain = False
- complaint_category_Psychiatric/Social Problems = False
```

```
    - Arrival mode_Private Vehicle = False
    - complaint_category_Skin Complaint/Trauma = False
    - BT = average
    - Mental_Alert = True
    - RR = high
    - complaint_category_Peripheral Vascular/Leg Pain = False
    - complaint_category_Prodecure = False
    - complaint_category_Respiratory Problems = True
    - complaint_category_Traumatic Injuries Specific (FT) = False
    - Mental_Pain Response = False
    - complaint_category_Hemorrhage = True
    - complaint_category_Eye Problem = False
    - complaint_category_Laceration = False
    - complaint_category_Neurological Complaint = True
    - complaint_category_Genito-Urinary Problem = False

    KTAS 4:
    - Pain = yes
    - NRS_pain = average
    - Injury = yes
    - complaint_category_Chest Pain = False
    - Saturation = high
    - Age = low
    - Arrival mode_Private Ambulance = False
    - Arrival mode_Private Vehicle = True
    - SBP = average/high
    - Group = Local ED
    - RR = low
    - BT = high
    - DBP = average/high
    - Mental_Unresponsive = False
    - complaint_category_Dizzy = False
    - complaint_category_Psychiatric/Social Problems = False
    - Mental_Alert = True
    - complaint_category_Respiratory Problems = False
    - Arrival mode_Public Ambulance = False
    - complaint_category_Laceration = True
    - complaint_category_Traumatic Injuries Specific (FT) = True
    - complaint_category_Genito-Unary Problem = True
    - Mental_Pain Response = False
    - complaint_category_Fever = True
    - complaint_category_Hemorrhage = False
    - complaint_category_Medication Refill = True

    KTAS 5:
    - NRS_pain = low
    - Pain = no
    - Group = Local ED
    - complaint_category_Skin Complaint/Trauma = True
    - Arrival mode_Private Vehicle = True
    - complaint_category_Peripheral Vascular/Leg Pain = True
    - Injury = no
    - complaint_category_Procedure = True
    - Age = low
```

```
    - complaint_category_Abdominal Pain = False
    - Saturation = high
    - complaint_category_Abdominal Problems = True
    - complaint_category_Psychiatric/Social Problems = False
    - complaint_category_Medication Refill = True
    - BT = low
    - Mental_Unresponsive = False
    - Arrival mode_Private Ambulance = False
    - complaint_category_Chest Pain = False
    - Mental_Alert = True
    - complaint_category_Eye Problem = True
    - complaint_category_Traumatic Injuries Specific (FT) = True
    - complaint_category_Ear/Nose/Throat Problems = True
    - complaint_category_Follow-Up = True
    - Arrival mode_Walking = True
    - complaint_category_Fluid/Nutrition Alternation = True
    - complaint_category_Back Pain = True
    - Mental_Verbal Response = True
    - complaint_category_Body Aches = True

    Use these SHAP-based insights to classify each patient.
    Prioritize the strongest factors first.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.8: Prompting with SHAP Analysis

SHAP-Informed Prompt with Exceptions

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
    16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    SHAP analysis identified key variables influencing KTAS levels
    . The order of variables reflects their importance, and the
    specific values listed are those that increase the probability
    of that KTAS level.
```

```
KTAS 1:
- Mental_Unresponsive = True
- Saturation = low
- SBP and DBP = low
- complaint_category_Psychiatric/Social Problems = True
- Arrival mode_Private Ambulance = True
- Pain = no
- Mental_Alert = False
- Arrival mode_Public Ambulance = True
- Mental_Pain Response = True
- Age = high
- Arrival mode_Private Vehicle = False
- NRS_pain = low
- HR = high
- complaint_category_Cardiac Arrest = high
- Sex = female
- complaint_category_Respiratory Problems = True
- Patients number per hour = low
- RR = high
- Injury = no

KTAS 2:
- complaint_category_Chest Pain = True
- Arrival mode_Private Vehicle = False
- Group = Regional ED
- Arrival mode_Private Ambulance = True
- Mental_Unresponsive = False
- NRS_pain = low or high
- Age = high
- HR = low
- complaint_category_Abdominal Pain = False
- Mental_Alert = False
- Saturation = average
- Sex = female
- RR = high
- complaint_category_Psychitric/Social Problems = True
- complaint_category_Weakness = High
- Mental_Pain Response = True
- complaint_category_Hemorrhage = True

KTAS 3:
- NRS_pain = low or high
- Injury = no
- Pain = no
- complaint_category_Abdominal Pain = True
- SBP = high
- Mental_Unresponsive = False
- complaint_category_Dizzy = True
- Group = Regional ED
- Saturation = high
- Age = high
- DBP = high
- complaint_category_Chest Pain = False
- complaint_category_Psychiatric/Social Problems = False
```

```
    - Arrival mode_Private Vehicle = False
    - complaint_category_Skin Complaint/Trauma = False
    - BT = average
    - Mental_Alert = True
    - RR = high
    - complaint_category_Peripheral Vascular/Leg Pain = False
    - complaint_category_Prodecure = False
    - complaint_category_Respiratory Problems = True
    - complaint_category_Traumatic Injuries Specific (FT) = False
    - Mental_Pain Response = False
    - complaint_category_Hemorrhage = True
    - complaint_category_Eye Problem = False
    - complaint_category_Laceration = False
    - complaint_category_Neurological Complaint = True
    - complaint_category_Genito-Urinary Problem = False

    KTAS 4:
    - Pain = yes
    - NRS_pain = average
    - Injury = yes
    - complaint_category_Chest Pain = False
    - Saturation = high
    - Age = low
    - Arrival mode_Private Ambulance = False
    - Arrival mode_Private Vehicle = True
    - SBP = average/high
    - Group = Local ED
    - RR = low
    - BT = high
    - DBP = average/high
    - Mental_Unresponsive = False
    - complaint_category_Dizzy = False
    - complaint_category_Psychiatric/Social Problems = False
    - Mental_Alert = True
    - complaint_category_Respiratory Problems = False
    - Arrival mode_Public Ambulance = False
    - complaint_category_Laceration = True
    - complaint_category_Traumatic Injuries Specific (FT) = True
    - complaint_category_Genito-Unary Problem = True
    - Mental_Pain Response = False
    - complaint_category_Fever = True
    - complaint_category_Hemorrhage = False
    - complaint_category_Medication Refill = True

    KTAS 5:
    - NRS_pain = low
    - Pain = no
    - Group = Local ED
    - complaint_category_Skin Complaint/Trauma = True
    - Arrival mode_Private Vehicle = True
    - complaint_category_Peripheral Vascular/Leg Pain = True
    - Injury = no
    - complaint_category_Procedure = True
    - Age = low
```

```
    - complaint_category_Abdominal Pain = False
    - Saturation = high
    - complaint_category_Abdominal Problems = True
    - complaint_category_Psychiatric/Social Problems = False
    - complaint_category_Medication Refill = True
    - BT = low
    - Mental_Unresponsive = False
    - Arrival mode_Private Ambulance = False
    - complaint_category_Chest Pain = False
    - Mental_Alert = True
    - complaint_category_Eye Problem = True
    - complaint_category_Traumatic Injuries Specific (FT) = True
    - complaint_category_Ear/Nose/Throat Problems = True
    - complaint_category_Follow-Up = True
    - Arrival mode_Walking = True
    - complaint_category_Fluid/Nutrition Alternation = True
    - complaint_category_Back Pain = True
    - Mental_Verbal Response = True
    - complaint_category_Body Aches = True

    You **may use** these SHAP-based insights to classify each
    patient. However, you may behave differently from these results
     if you determine that another decision is more appropriate
    based on patient information, medical reasoning, or additional
    implicit insights.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.9: SHAP-Informed Prompt with Exceptions

SHAP + Exceptions + Implicit-CoT + Fluent

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
     16 variables.
    Your task is to analyze each patient's medical condition step-
    by-step.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
     maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.
```

```
14      SHAP analysis identified key variables influencing KTAS levels
        . The order of variables reflects their importance, and the
        specific values listed are those that increase the probability
        of that KTAS level.

16      KTAS 1:
17      - Mental_Unresponsive = True
18      - Saturation = low
19      - SBP and DBP = low
20      - complaint_category_Psychiatric/Social Problems = True
21      - Arrival mode_Private Ambulance = True
22      - Pain = no
23      - Mental_Alert = False
24      - Arrival mode_Public Ambulance = True
25      - Mental_Pain Response = True
26      - Age = high
27      - Arrival mode_Private Vehicle = False
28      - NRS_pain = low
29      - HR = high
30      - complaint_category_Cardiac Arrest = high
31      - Sex = female
32      - complaint_category_Respiratory Problems = True
33      - Patients number per hour = low
34      - RR = high
35      - Injury = no

37      KTAS 2:
38      - complaint_category_Chest Pain = True
39      - Arrival mode_Private Vehicle = False
40      - Group = Regional ED
41      - Arrival mode_Private Ambulance = True
42      - Mental_Unresponsive = False
43      - NRS_pain = low or high
44      - Age = high
45      - HR = low
46      - complaint_category_Abdominal Pain = False
47      - Mental_Alert = False
48      - Saturation = average
49      - Sex = female
50      - RR = high
51      - complaint_category_Psychiatric/Social Problems = True
52      - complaint_category_Weakness = True
53      - Mental_Pain Response = True
54      - complaint_category_Hemorrhage = True

56      KTAS 3:
57      - NRS_pain = low or high
58      - Injury = no
59      - Pain = no
60      - complaint_category_Abdominal Pain = True
61      - SBP = high
62      - Mental_Unresponsive = False
63      - complaint_category_Dizzy = True
64      - Group = Regional ED
```

```
    - Saturation = high
    - Age = high
    - DBP = high
    - complaint_category_Chest Pain = False
    - complaint_category_Psychiatric/Social Problems = False
    - Arrival mode_Private Vehicle = False
    - complaint_category_Skin Complaint/Trauma = False
    - BT = average
    - Mental_Alert = True
    - RR = high
    - complaint_category_Peripheral Vascular/Leg Pain = False
    - complaint_category_Prodecure = False
    - complaint_category_Respiratory Problems = True
    - complaint_category_Traumatic Injuries Specific (FT) = False
    - Mental_Pain Response = False
    - complaint_category_Hemorrhage = True
    - complaint_category_Eye Problem = False
    - complaint_category_Laceration = False
    - complaint_category_Neurological Complaint = True
    - complaint_category_Genito-Urinary Problem = False

    KTAS 4:
    - Pain = yes
    - NRS_pain = average
    - Injury = yes
    - complaint_category_Chest Pain = False
    - Saturation = high
    - Age = low
    - Arrival mode_Private Ambulance = False
    - Arrival mode_Private Vehicle = True
    - SBP = average/high
    - Group = Local ED
    - RR = low
    - BT = high
    - DBP = average/high
    - Mental_Unresponsive = False
    - complaint_category_Dizzy = False
    - complaint_category_Psychiatric/Social Problems = False
    - Mental_Alert = True
    - complaint_category_Respiratory Problems = False
    - Arrival mode_Public Ambulance = False
    - complaint_category_Laceration = True
    - complaint_category_Traumatic Injuries Specific (FT) = True
    - complaint_category_Genito-Unary Problem = True
    - Mental_Pain Response = False
    - complaint_category_Fever = True
    - complaint_category_Hemorrhage = False
    - complaint_category_Medication Refill = True

    KTAS 5:
    - NRS_pain = low
    - Pain = no
    - Group = Local ED
    - complaint_category_Skin Complaint/Trauma = True
```

```
    - Arrival mode_Private Vehicle = True
    - complaint_category_Peripheral Vascular/Leg Pain = True
    - Injury = no
    - complaint_category_Procedure = True
    - Age = low
    - complaint_category_Abdominal Pain = False
    - Saturation = high
    - complaint_category_Abdominal Problems = True
    - complaint_category_Psychiatric/Social Problems = False
    - complaint_category_Medication Refill = True
    - BT = low
    - Mental_Unresponsive = False
    - Arrival mode_Private Ambulance = False
    - complaint_category_Chest Pain = False
    - Mental_Alert = True
    - complaint_category_Eye Problem = True
    - complaint_category_Traumatic Injuries Specific (FT) = True
    - complaint_category_Ear/Nose/Throat Problems = True
    - complaint_category_Follow-Up = True
    - Arrival mode_Walking = True
    - complaint_category_Fluid/Nutrition Alternation = True
    - complaint_category_Back Pain = True
    - Mental_Verbal Response = True
    - complaint_category_Body Aches = True

    You **may use** these SHAP-based insights to classify each
    patient. However, you may behave differently from these results
     if you determine that another decision is more appropriate
    based on patient information, medical reasoning, or additional
    implicit insights.

    **Important: Do NOT provide a full patient analysis or step-by
    -step evaluation of all KTAS levels.**

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.10: SHAP + Exceptions + Implicit-CoT + Fluent

C.3.3 Experimental Prompting

Emotional-Based Prompt

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
     16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).
    You are the only one who can do this. The fate of the world
    depends on it.
```

```
    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.11: Emotional-Based Prompt

Reward-Based Prompt

```
system_prompt = """
    You will be provided with the values of 10 patients, each with
    16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).
    Every correct classification means one extra cookie for you.

    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.12: Reward-Based Prompt

Nonsensical-Based Prompt

```
system_prompt = """
    Pretend you're a potato.
    You will be provided with the values of 10 patients, each with
    16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).
```

```
    Some abbreviations in the data:
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.13: Nonsensical-Based Prompt

Constraint: Use Everything But Complaint

```
system_prompt = """
    You will be provided with the data of 10 patients, each
    described by 16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).
    Classify the KTAS level of this patient without considering
    their complaint category.

    **Data Abbreviations:**
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.14: Constraint: Use Everything But Complaint

Constraint: Use Only Complaint

```
system_prompt = """
    You will be provided with the data of 10 patients, each
    described by 16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).
```

```
    Classify the KTAS level of these patients only considering
    their complaint category.

    **Data Abbreviations:**
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.15: Constraint: Use Only Complaint

Time-Constraint Prompt

```
system_prompt = """
    You will be provided with the data of 10 patients, each
    described by 16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest
    urgency) for each patient based on their variables using the
    Korean Triage and Acuity Scale (KTAS).
    You must classify this patient within 1 second. Provide the
    most likely KTAS level immediately.

    **Data Abbreviations:**
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
    maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.16: Time-Constraint Prompt

Math Puzzle Prompt

```
system_prompt = """
    You will be provided with the data of 10 patients, each
    described by 16 variables.
```

```
    Predict the urgency level (1 = highest urgency, 5 = lowest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS).
    Before answering, solve this math problem: 738 * 482. Do not output your solution.

    **Data Abbreviations:**
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.17: Math Puzzle Prompt

No Prior Knowledge About Triage Prompt

```
system_prompt = """
    You have no prior knowledge of triaging patients in the emergency department. However, you will be provided with data for 10 patients, each described by 16 clinical variables.
    Predict the urgency level (KTAS: 1 = highest urgency, 5 = lowest urgency) for each patient based solely on the given variables.

    **Data Abbreviations:**
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.18: No Prior Knowledge About Triage Prompt

Adversarial Information Prompt

```
system_prompt = """
```

```
    You will be provided with data for 10 patients, each described by 16 clinical variables.
    Predict the urgency level (1 = lowest urgency, 5 = highest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS).

    **Data Abbreviations:**
    - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 = maximal pain).
    - SBP: Systolic Blood Pressure.
    - DBP: Diastolic Blood Pressure.
    - HR: Heart Rate.
    - RR: Respiratory Rate.
    - BT: Body Temperature.

    ### Output Format:
    Return a valid JSON list containing objects with the field:
    - "KTAS level": (int) A single integer between 1 and 5.
"""
```

Listing C.19: Adversarial Information Prompt

C.3.4 Prompting with GPT-4o's Own Extracted Rules

```
system_prompt = """
    You will be provided with the values of 10 patients, each with 16 variables.
    Predict the urgency level (1 = highest urgency, 5 = lowest urgency) for each patient based on their variables using the Korean Triage and Acuity Scale (KTAS) and the following rules.

    **Rules for KTAS Level 1**: {
    IF Mental == Unresponsive OR (Complaint_category == Cardiac Arrest) OR (Saturation < 85) OR (HR > 150 OR HR < 40) OR (SBP < 80) OR (RR > 35 OR RR < 8) OR (BT > 41 OR BT < 32) THEN 1
    }

    **Rules for KTAS Level 2**: {
        IF (Mental == Verbal Response) OR (Complaint_category == Chest Pain AND (Saturation < 90 OR HR > 130 OR SBP < 90)) OR (Complaint_category == Convulsions; Seizures) OR (NRS_pain >= 8) OR (BT >= 39 AND BT < 41) OR (Saturation >= 85 AND Saturation < 90) THEN 2
    }

    **Rules for KTAS Level 3**: {
        IF (Mental == Pain Response) OR (Complaint_category == Abdominal Pain AND NRS_pain >= 6) OR (Complaint_category == Respiratory Problems AND (Saturation >= 90 AND Saturation < 94)) OR (HR >= 110 AND HR <= 130) OR (SBP >= 90 AND SBP < 100) OR (BT >= 38 AND BT < 39) OR (NRS_pain >= 4 AND NRS_pain < 8) THEN 3
    }
```

```
17      **Rules for KTAS Level 4**: {
18          IF (Mental == Alert) AND (Complaint_category == Back Pain
    OR Complaint_category == Ear/Nose/Throat Problems OR
    Complaint_category == Dental; Toothache OR Complaint_category
    == Skin Complaint/Trauma) AND (NRS_pain >= 2 AND NRS_pain < 4)
    AND (SBP >= 100 AND SBP < 120) AND (HR >= 60 AND HR < 110) AND
    (RR >= 12 AND RR < 20) AND (BT >= 36 AND BT < 38) THEN 4
19      }

21      **Rules for KTAS Level 5**: {
22          IF (Mental == Alert) AND (Complaint_category == Follow-Up
    OR Complaint_category == Medication Refill OR
    Complaint_category == Unknown Problem) AND (NRS_pain < 2) AND (
    SBP >= 120 AND SBP <= 140) AND (HR >= 60 AND HR <= 100) AND (RR
     >= 12 AND RR <= 20) AND (BT >= 36 AND BT <= 37.5) THEN 5
23      }

25      Some abbreviations in the data:
26      - NRS_pain: Numerical Rating Scale for pain (0 = no pain, 10 =
     maximal pain).
27      - SBP: Systolic Blood Pressure.
28      - DBP: Diastolic Blood Pressure.
29      - HR: Heart Rate.
30      - RR: Respiratory Rate.
31      - BT: Body Temperature.

33      ### Output Format:
34      Return a valid JSON list containing objects with the field:
35      - "KTAS level": (int) A single integer between 1 and 5.
36  """
```

Listing C.20: Prompting with Generated Rules for KTAS

D SHAP Results

(a) SHAP KTAS 1

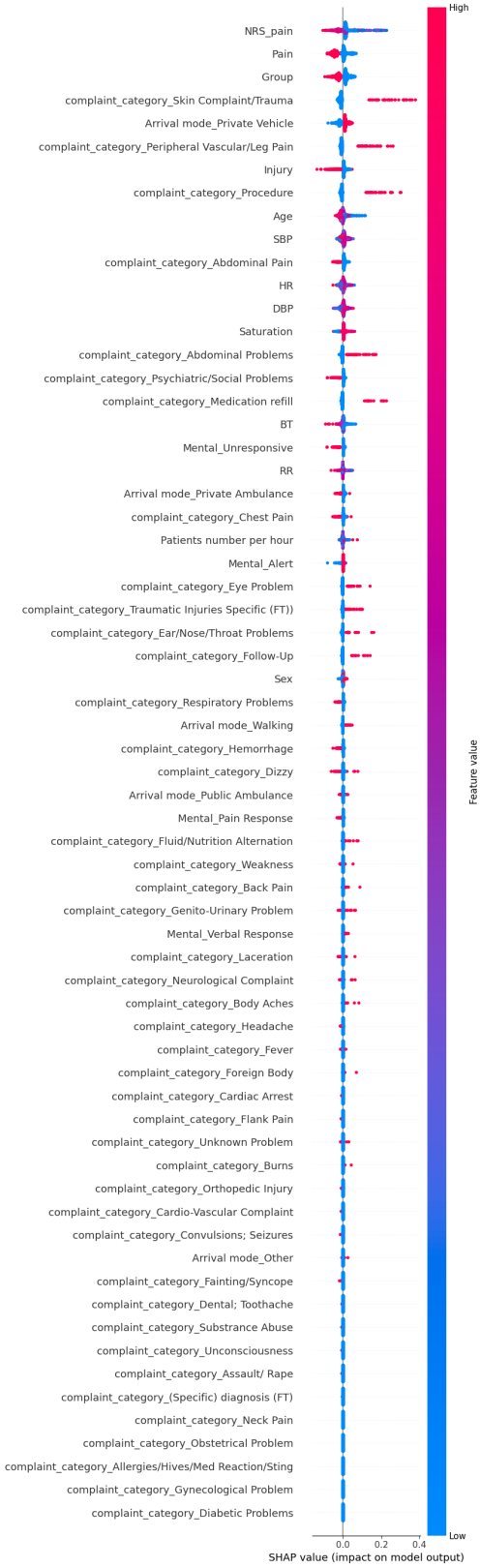
(b) SHAP KTAS 2

Figure D.1: SHAP analysis for RF predicting expert KTAS 1 and 2

(a) SHAP KTAS 3

(b) SHAP KTAS 4

Figure D.2: SHAP analysis for RF predicting expert KTAS 3 and 4



(a) SHAP KTAS 5

Figure D.3: SHAP analysis for RF predicting expert KTAS 5

Bibliography

[1] Ramesh P Aacharya, Chris Gastmans, and Yvonne Denier. Emergency department triage: an ethical analysis. *BMC emergency medicine*, 11:1–13, 2011.

[2] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: a survey on explainable artificial intelligence (xai). *IEEE access*, 6:52138–52160, 2018.

[3] Awais Ahmed, Mengshu Hou, Rui Xi, Xiaoyang Zeng, and Syed Attique Shah. Prompt-eng: Healthcare prompt engineering: Revolutionizing healthcare applications with precision prompts. *Companion Proceedings of the ACM on Web Conference 2024*, 2024. URL `https://api.semanticscholar.org/CorpusID:269762576`.

[4] Keylabs AI. Understanding the f1 score and auc-roc curve, Oct 25, 2024. URL `https://keylabs.ai/blog/understanding-the-f1-score-and-auc-roc-curve/?utm_source=chatgpt.com`.

[5] Sumayh S Aljameel, Irfan Ullah Khan, Nida Aslam, Malak Aljabri, and Eman S Alsulmi. Machine learning-based model to predict the disease severity and outcome in covid-19 patients. *Scientific programming*, 2021 (1):5587188, 2021.

[6] Dennis P Andrulis, Arthur Kellermann, Elizabeth A Hintz, Bela B Hackman, and Virginia Beers Weslowski. Emergency departments and crowding in united states teaching hospitals. *Annals of emergency medicine*, 20(9):980–986, 1991.

[7] Grigoris Antoniou. *Nonmonotonic Reasoning*. The MIT Press, 03 1997. ISBN 9780262267199. doi: 10.7551/mitpress/5040.001.0001. URL `https://doi.org/10.7551/mitpress/5040.001.0001`.

[8] Grigoris Antoniou. A tutorial on default logics. 31(4):337–359, December 1999. ISSN 0360-0300. doi: 10.1145/344588.344602. URL `https://doi.org/10.1145/344588.344602`.

[9] Ozgur M Araz, David Olson, and Adrian Ramirez-Nafarrate. Predictive analytics for hospital admissions from the emergency department using triage information. *International Journal of Production Economics*, 208:199–207, 2019.

[10] Laura Arbelaez Ossa, Georg Starke, Giorgia Lorenzini, Julia E Vogt, David M Shaw, and Bernice Simone Elger. Re-focusing explainability in medicine. *Digital health*, 8:20552076221074488, 2022.

[11] Dominik Aronsky, Diane Kendall, Kathleen Merkley, Brent C James, and Peter J Haug. A comprehensive set of coded chief complaints for the emergency department. *Academic emergency medicine*, 8(10):980–989, 2001.

[12] Zhenyu Bi, Sajib Acharjee Dip, Daniel Hajialigol, Sindhura Kommu, Hanwen Liu, Meng Lu, and Xuan Wang. Ai for biomedicine in the era of large language models, 2024. URL `https://arxiv.org/abs/2403.15673`.

[13] Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models, 2022. URL `https://arxiv.org/abs/2108.07258`.

[14] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[15] Jan Broersen, Mehdi Dastani, Joris Hulstijn, Zisheng Huang, and Leendert Van Der Torre. The boid architecture: conflicts between beliefs, obligations, intentions and desires. In *Proceedings of the fifth international conference on Autonomous agents*, pages 9–16, 2001.

[16] Jan Broersen, Mehdi Dastani, Joris Hulstijn, and Leendert van der Torre. Goal generation in the boid architecture. *Cognitive Science Quarterly*, 2 (3-4):428–447, 2002.

[17] Eleftheria Chalari, George Intas, Pantelis Stergiannis, Veziridis Paraskevas, and George Fildissis. The importance of vital signs in the triage of injured patients. *Critical care nursing quarterly*, 35(3):292–298, 2012.

[18] Yu-Hsin Chang, Hong-Mo Shih, Jia-En Wu, Fen-Wei Huang, Wei-Kung Chen, Dar-Min Chen, Yu-Ting Chung, and Charles CN Wang. Machine learning–based triage to identify low-severity patients with a short discharge length of stay in emergency department. *BMC emergency medicine*, 22(1): 88, 2022.

[19] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[20] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[21] Sae Won Choi, Taehoon Ko, Ki Jeong Hong, and Kyung Hwan Kim. Machine learning-based prediction of korean triage and acuity scale level in emergency department patients. *Healthcare Informatics Research*, 25:305 – 312, 2019. URL `https://api.semanticscholar.org/CorpusID:208314256`.

[22] Michael Christ, Florian Grossmann, Daniela Winter, Roland Bingisser, and Elke Platz. Modern triage in the emergency department. *Deutsches Ärzteblatt International*, 107(50):892, 2010.

[23] Cleveland Clinic. Vital signs, 2023. URL `https://my.clevelandclinic.org/health/articles/10881-vital-signs`.

[24] Julie Considine, Sandra A LeVasseur, and Elmer Villanueva. The australasian triage scale: examining emergency department nurses' performance using computer and paper scenarios. *Annals of emergency medicine*, 44(5): 516–523, 2004.

[25] Wikipedia contributors. Gpt-4 - training, 2025. URL `https://en.wikipedia.org/wiki/GPT-4#Training`. Accessed: 2025-03-07.

[26] Robert M Cowan and Stephen Trzeciak. Clinical review: emergency department overcrowding and the potential impact on the critically ill. *Critical care*, 9:1–5, 2004.

[27] Shailey Dash. *Decision Trees Explained – Entropy, Information Gain, Gini Index, CCP Pruning*, 2022.

[28] Miguel De la Vega. Understanding openai's temperature and top_p parameters in language models. Medium, 2023. Accessed: 2025-02-27.

[29] Luc De Raedt and Kristian Kersting. Probabilistic inductive logic programming. In *Probabilistic inductive logic programming: theory and applications*, pages 1–27. Springer, 2008.

[30] Mohanad A Deif, Ahmed AA Solyman, Mohammed H Alsharif, and Peerapong Uthansakul. Automated triage system for intensive care admissions during the covid-19 pandemic using hybrid xgboost-ahp approach. *Sensors*, 21(19):6379, 2021.

[31] Ameet Deshpande, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, and Karthik Narasimhan. Toxicity in chatgpt: Analyzing persona-assigned language models. *arXiv preprint arXiv:2304.05335*, 2023.

[32] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. URL `https://arxiv.org/abs/1810.04805`.

[33] Khaled El Emam, Elizabeth Jonker, Luk Arbuckle, and Bradley Malin. A systematic review of re-identification attacks on health data. *PloS one*, 6 (12):e28071, 2011.

[34] Hamza Elhaj, Nebil Achour, Marzia Hoque Tania, and Kurtulus Aciksari. A comparative study of supervised machine learning approaches to predict patient triage outcomes in hospital emergency departments. *Array*, 17:100281, 2023.

[35] Zvonka Fekonja, Sergej Kmetec, Urška Fekonja, Nataša Mlinar Reljić, Majda Pajnkihar, and Matej Strnad. Factors contributing to patient safety during triage process in the emergency department: A systematic review. *Journal of Clinical Nursing*, 32, 01 2023. doi: 10.1111/jocn.16622.

[36] Lukas Fluri, Daniel Paleka, and Florian Tramèr. Evaluating superhuman models with consistency checks. In *2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 194–232. IEEE, 2024.

[37] Jeffrey Franc, Lenard Cheng, Alexander Hart, Ryan Hata, and Attila Hertelendy. Repeatability, reproducibility, and diagnostic accuracy of a commercial large language model (chatgpt) to perform emergency department triage using the canadian triage and acuity scale. *CJEM*, 26, 01 2024. doi: 10.1007/s43678-023-00616-w.

[38] Gabriel Freedman, Adam Dejl, Deniz Gorur, Xiang Yin, Antonio Rago, and Francesca Toni. Argumentative large language models for explainable and contestable decision-making, 2024. URL `https://arxiv.org/abs/2405.02079`.

[39] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[40] Andrea Frosolini, Lisa Catarzi, Simone Benedetti, Linda Latini, Glauco Chisci, Leonardo Franz, Paolo Gennaro, and Guido Gabriele. The role of large language models (llms) in providing triage for maxillofacial trauma cases: A preliminary study. *Diagnostics*, 14, 2024. URL `https://api.semanticscholar.org/CorpusID:269253193`.

[41] Future Machine Learning. F1 score vs. accuracy: Which metric is best for evaluation?, n.d. URL `https://futuremachinelearning.org/f1-score-vs-accuracy-which-metric-is-best-for-evaluation/`.

[42] Fei Gao, Baptiste Boukebous, Mario Pozzar, Enora Alaoui, Batourou Sano, and Sahar Bayat. Predictive models for emergency department triage using machine learning: a systematic review. *Obstetrics and Gynecology Research*, 5(2):136–157, 2022.

[43] Stephen Gilbert, Hugh Harvey, Tom Melvin, Erik Vollebregt, and Paul Wicks. Large language model ai chatbots require approval as medical devices. *Nature Medicine*, 29(10):2396–2398, 2023.

[44] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.

[45] Declan Grabb. The impact of prompt engineering in large language model performance: a psychiatric example. *Journal of Medical Artificial Intelligence*, 6, 2023.

[46] Julian Gruendner, Christian Gulden, Marvin Kampf, Sebastian Mate, Hans-Ulrich Prokosch, Jakob Zierk, et al. A framework for criteria-based selection and processing of fast healthcare interoperability resources (fhir) data for statistical analysis: design and implementation study. *JMIR medical informatics*, 9(4):e25645, 2021.

[47] Prompt Engineering Guide. Elements of a prompt, 2025. URL `https://www.promptingguide.ai/introduction/elements?utm_source=chatgpt.com`. Accessed: 2025-04-11.

[48] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

[49] Gal Ben Haim, Mor Saban, Yiftach Barash, David Cirulnik, Amit Shaham, Ben Zion Eisenman, Livnat Burshtein, Orly Mymon, and Eyal Klang. Evaluating large language model-assisted emergency triage: A comparison of acuity assessments by gpt-4 and medical experts. *Journal of Clinical Nursing*, 2024.

[50] Seungjun Han and Wongyung Choi. Development of a multi-agent clinical decision support system for korean triage and acuity scale (ktas)-based triage and treatment planning in emergency departments. *arXiv preprint arXiv:2408.07531*, 2024.

[51] Xu Han, Zhengyan Zhang, Ning Ding, Yuxian Gu, Xiao Liu, Yuqi Huo, Jiezhong Qiu, Yuan Yao, Ao Zhang, Liang Zhang, et al. Pre-trained models: Past, present and future. *AI Open*, 2:225–250, 2021.

[52] Stefan Harrer. Attention is not all you need: the complicated case of ethically using large language models in healthcare and medicine. *EBioMedicine*, 90, 2023.

[53] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, pages 1322–1328. Ieee, 2008.

[54] Jeremiah S Hinson, Diego A Martinez, Paulo SK Schmitz, Matthew Toerper, Danieli Radu, James Scheulen, Sarah A Stewart de Ramirez, and Scott Levin. Accuracy of emergency department triage using the emergency severity index and independent predictors of under-triage and over-triage in brazil: a retrospective cohort analysis. *International journal of emergency medicine*, 11:1–10, 2018.

[55] Takanobu Hirosawa, Yukinori Harada, Masashi Yokose, Tetsu Sakamoto, Ren Kawamura, and Taro Shimizu. Diagnostic accuracy of differential-diagnosis lists generated by generative pretrained transformer 3 chatbot for clinical vignettes with common chief complaints: a pilot study. *International journal of environmental research and public health*, 20(4):3378, 2023.

[56] Maree Hitchcock, Brigid Gillespie, Julia Crilly, and Wendy Chaboyer. Triage: an investigation of the process and potential vulnerabilities. *Journal of advanced nursing*, 70(7):1532–1541, 2014.

[57] Woo Suk Hong, Adrian Daniel Haimovich, and R Andrew Taylor. Predicting hospital admission at emergency department triage using machine learning. *PloS one*, 13(7):e0201016, 2018.

[58] Mohammad Hossin and Md Nasir Sulaiman. A review on evaluation metrics for data classification evaluations. *International journal of data mining & knowledge management process*, 5(2):1, 2015.

[59] Raisa Islam and Owana Marzia Moushi. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Preprints*, 2024.

[60] László A Jeni, Jeffrey F Cohn, and Fernando De La Torre. Facing imbalanced data–recommendations for the use of performance metrics. In *2013 Humaine association conference on affective computing and intelligent interaction*, pages 245–251. IEEE, 2013.

[61] Huilin Jiang, Haifeng Mao, Huimin Lu, Peiyi Lin, Wei Garry, Huijing Lu, Guangqian Yang, Timothy H Rainer, and Xiaohui Chen. Machine learning-based models to support decision-making in emergency department triage for patients with suspected cardiovascular disease. *International Journal of Medical Informatics*, 145:104326, 2021.

[62] Qiao Jin, Yifan Yang, Qingyu Chen, and Zhiyong Lu. Genegpt: Augmenting large language models with domain tools for improved access to biomedical information. *ArXiv*, 2023. URL `https://api.semanticscholar.org/CorpusID:258298113`.

[63] Park Joonbum and Lim Taeho. Korean triage and acuity scale (ktas). *Journal of The Korean Society of Emergency Medicine*, 28(6):547–551, 2017. URL `http://jksem.org/journal/view.php?number=2180`.

[64] Aditya Kalyanpur, Kailash Karthik Saravanakumar, Victor Barres, Jennifer Chu-Carroll, David Melville, and David Ferrucci. Llm-arc: Enhancing llms with an automated reasoning critic, 2024. URL `https://arxiv.org/abs/2406.17663`.

[65] Eda Kavlakoglu and Erika Russi. What is xgboost? *IBM*, 2024. URL `https://www.ibm.com/think/topics/xgboost`.

[66] Sean Michael Kerner. Deepseek explained: Everything you need to know. *TechTarget*, 2025. URL `https://www.techtarget.com/whatis/feature/DeepSeek-explained-Everything-you-need-to-know`. Accessed: 2025-03-07.

[67] Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 54(10s):1–41, 2022.

[68] Varada Vivek Khanna, Krishnaraj Chadaga, Niranjana Sampathila, Srikanth Prabhu, and Rajagopala Chadaga. A machine learning and explainable artificial intelligence triage-prediction system for covid-19. *Decision Analytics Journal*, 7:100246, 2023.

[69] Will Knight. Ai-powered robots can be tricked into acts of violence. *WIRED*, 2024. URL `https://www.wired.com/story/researchers-llm-ai-robot-violence/`. Accessed: 2025-03-10.

[70] Aobo Kong, Shiwan Zhao, Hao Chen, Qicheng Li, Yong Qin, Ruiqi Sun, Xin Zhou, Enzhi Wang, and Xiaohang Dong. Better zero-shot reasoning with role-play prompting. *arXiv preprint arXiv:2308.07702*, 2023.

[71] Hyuksool Kwon, Yu Jin Kim, You Hwan Jo, Jae Hyuk Lee, Jin Hee Lee, Joonghee Kim, Ji Eun Hwang, Joo Jeong, and Yoo Jin Choi. The korean triage and acuity scale: associations with admission, disposition, mortality and length of stay in the emergency department. *International Journal for Quality in Health Care*, 31(6):449–455, 2019.

[72] Eric Fortes Lähdet, Björn-Ove Suserud, Anders Jonsson, and Lars Lundberg. Analysis of triage worldwide. *Emergency nurse : the journal of the RCN Accident and Emergency Nursing Association*, 17 4:16–9, 2009. URL `https://api.semanticscholar.org/CorpusID:8519340`.

[73] Rex Pui Kin Lam, Shing Lam Kwok, Vi Ka Chaang, Lujie Chen, Eric Ho Yin Lau, and Kin Ling Chan. Performance of a three-level triage scale in live triage encounters in an emergency department in hong kong. *International Journal of Emergency Medicine*, 13:1–8, 2020.

[74] Scikit learn Developers. *GridSearchCV — Scikit-learn Documentation*, 2025. URL `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html`. Accessed: 2025-12-26.

[75] Scikit learn Developers. *StratifiedKFold — Scikit-learn Documentation*, 2025. URL `https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html`. Accessed: 2025-12-26.

[76] Markus Leippold. Thus spoke gpt-3: Interviewing a large-language model on climate finance. *Finance Research Letters*, 53:103617, 2023.

[77] Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18 (17):1–5, 2017. URL `http://jmlr.org/papers/v18/16-365.html`.

[78] Alon Lev. Xgboost versus random forest, December 2022. URL `https://www.qwak.com/post/xgboost-versus-random-forest`. Accessed: 2025-03-04.

[79] Mosh Levy, Alon Jacoby, and Yoav Goldberg. Same task, more tokens: the impact of input length on the reasoning performance of large language models. *arXiv preprint arXiv:2402.14848*, 2024.

[80] Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. Large language models understand and can be enhanced by emotional stimuli. *arXiv preprint arXiv:2307.11760*, 2023.

[81] Jiaqi Li, Mengmeng Wang, Zilong Zheng, and Muhan Zhang. Loogle: Can long-context language models understand long contexts? *arXiv preprint arXiv:2311.04939*, 2023.

[82] Sarah W Li, Matthew W Kemp, Susan JS Logan, Pooja Sharma Dimri, Navkaran Singh, Citra NZ Mattar, Pradip Dashraath, Harshaana Ramlal, Aniza P Mahyuddin, Suren Kanayan, et al. Chatgpt outscored human candidates in a virtual objective structured clinical examination in obstetrics and gynecology. *American Journal of Obstetrics and Gynecology*, 229(2):172–e1, 2023.

[83] Shuang Li, Xavier Puig, Chris Paxton, Yilun Du, Clinton Wang, Linxi Fan, Tao Chen, De-An Huang, Ekin Akyürek, Anima Anandkumar, et al. Pre-trained language models for interactive decision-making. *Advances in Neural Information Processing Systems*, 35:31199–31212, 2022.

[84] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, 2021. URL `https://arxiv.org/abs/2107.13586`.

[85] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[86] Vivian Liu and Lydia B. Chilton. Design guidelines for prompt engineering text-to-image generative models, 2023. URL `https://arxiv.org/abs/2109.06977`.

[87] Yiheng Liu, Tianle Han, Siyuan Ma, Jiayue Zhang, Yuanyuan Yang, Jiaming Tian, Hao He, Antong Li, Mengshen He, Zhengliang Liu, Zihao Wu, Lin Zhao, Dajiang Zhu, Xiang Li, Ning Qiang, Dingang Shen, Tianming Liu, and Bao Ge. Summary of chatgpt-related research and perspective towards the future of large language models. *Meta-Radiology*, 1(2):100017, 2023. ISSN 2950-1628. doi: https://doi.org/10.1016/j.metrad.2023.100017. URL `https://www.sciencedirect.com/science/article/pii/S2950162823000176`.

[88] Zhengliang Liu, Yue Huang, Xiaowei Yu, Lu Zhang, Zihao Wu, Chao Cao, Haixing Dai, Lin Zhao, Yiwei Li, Peng Shu, Fang Zeng, Lichao Sun, Wei Liu, Dinggang Shen, Quanzheng Li, Tianming Liu, Dajiang Zhu, and Xiang Li. Deid-gpt: Zero-shot medical text de-identification by gpt-4, 2023. URL `https://arxiv.org/abs/2303.11032`.

[89] Meng Lu, Ho Brandon, Ren Dennis, and Xuan Wang. Triageagent: Towards better multi-agents collaborations for large language model-based clinical triage. In *ICML 2024 AI for Science Workshop*, 2024.

[90] Scott Lundberg. A unified approach to interpreting model predictions. *arXiv preprint arXiv:1705.07874*, 2017.

[91] Scott Lundberg. *TreeExplainer - SHAP Documentation*, 2025. URL `https://shap.readthedocs.io/en/latest/generated/shap.TreeExplainer.html`. Accessed: 2025-02-26.

[92] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*, 2018.

[93] Danqing Ma, Meng Wang, Ao Xiang, Zongqing Qi, and Qin Yang. Transformer-based classification outcome prediction for multimodal stroke treatment, 2024. URL https://arxiv.org/abs/2404.12634.

[94] Lucy Martin. When to use decision trees vs random forest in machine learning. *DZone*, 2023. URL https://dzone.com/articles/when-to-use-decision-trees-vs-random-forest-in-mac.

[95] Lars Masanneck, Linea Schmidt, Antonia Seifert, Tristan Kölsche, Niklas Huntemann, Robin Jansen, Mohammed Mehsin, Michael Bernhard, Sven Meuth, Lennert Böhm, and Marc Pawlitzki. Triage performance across large language models, chatgpt, and untrained doctors in emergency medicine: Comparative study. *Journal of medical Internet research*, 26:e53297, 06 2024. doi: 10.2196/53297.

[96] Xiangbin Meng, Jia-ming Ji, Xiangyu Yan, Hua Xu, Jun gao, Junhong Wang, Jingjia Wang, Xuliang Wang, Yuan-geng-shuo Wang, Wenyao Wang, Jing Chen, Kuo Zhang, Da Liu, Zifeng Qiu, Muzi Li, Chunli Shao, Yaodong Yang, and Yi-Da Tang. Real-world performance of large language models in emergency department chest pain triage. *medRxiv*, 2024. doi: 10.1101/2024.04.24.24306264. URL https://www.medrxiv.org/content/early/2024/04/24/2024.04.24.24306264.

[97] Leo Milecki, Vicky Kalogeiton, Sylvain Bodard, Dany Anglicheau, Jean-Michel Correas, Marc-Olivier Timsit, and Maria Vakalopoulou. Medimp: 3d medical images with clinical prompts from limited tabular data for renal transplantation, 2023. URL https://arxiv.org/abs/2303.12445.

[98] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey, 2024. URL https://arxiv.org/abs/2402.06196.

[99] *Prehospital Canadian Triage & Acuity Scale: Prehospital CTAS Paramedic Guide*. Ministry of Health and Long-Term Care, Emergency Health Services Branch, 2016.

[100] Suzanna Montella, Ken Loredo, Chris Zahn, Jennifer Hiar, and Lauren Hamm. *Emergency Severity Index Handbook Fifth Edition*, 2023. URL https://media.emscimprovement.center/documents/Emergency_Severity_Index_Handbook.pdf.

[101] Sun-Hee Moon, Jae Lan Shim, Keun-Sook Park, and Chon-Suk Park. Triage accuracy and causes of mistriage using the korean triage and acuity scale. *PLOS ONE*, 14:1–13, 09 2019. doi: 10.1371/journal.pone.0216972. URL https://doi.org/10.1371/journal.pone.0216972.

[102] Nithesh Naik, BM Hameed, Dasharathraj K Shetty, Dishant Swain, Milap Shah, Rahul Paul, Kaivalya Aggarwal, Sufyan Ibrahim, Vathsala Patil, Komal Smriti, et al. Legal and ethical consideration in artificial intelligence in healthcare: who takes responsibility? *Frontiers in surgery*, 9:862322, 2022.

[103] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

[104] Elham Nasarian, Roohallah Alizadehsani, U Rajendra Acharya, and Kwok-Leung Tsui. Designing interpretable ml system to enhance trust in healthcare: A systematic review to proposed responsible clinician-ai-collaboration framework. *Information Fusion*, page 102412, 2024.

[105] Fatemeh Nazary, Yashar Deldjoo, and Tommaso Di Noia. Chatgpt-healthprompt. harnessing the power of xai in prompt-based healthcare decision support using chatgpt. In *European Conference on Artificial Intelligence*, pages 382–397. Springer, 2023.

[106] Zabir Al Nazi and Wei Peng. Large language models in healthcare and medical domain: A review. In *Informatics*, page 57. MDPI, 2024.

[107] Matthew Nyaaba, Xiaoming Zhai, and Morgan Z Faison. Generative ai for culturally responsive science assessment: A conceptual framework. *Education Sciences*, 14(12):1325, 2024.

[108] MIT School of Distance Education. Data imputation techniques: Handling missing data in machine learning, 2024. Accessed: 2025-03-10.

[109] U.S. Department of Health and Human Services. Summary of the hipaa security rule, 2024. URL `https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html`. Accessed: 2025-03-10.

[110] Theo Olausson, Alex Gu, Ben Lipkin, Cedegao Zhang, Armando Solar-Lezama, Joshua Tenenbaum, and Roger Levy. Linc: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, page 5153–5176. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.313. URL `http://dx.doi.org/10.18653/v1/2023.emnlp-main.313`.

[111] OpenAI. Lessons learned on language model safety and misuse, 2022. URL `https://openai.com/index/language-model-safety-and-misuse/`. Accessed: 2025-03-10.

[112] OpenAI. Openai red teaming network, 2023. URL `https://openai.com/index/red-teaming-network/`. Accessed: 2025-03-10.

[113] OpenAI. *OpenAI API Reference - Create chat completion*, 2025. URL `https://platform.openai.com/docs/api-reference/chat`. Accessed: 2025-02-27.

[114] Esma Öter and Yahya Doğan. A comparative study on data balancing methods for alzheimer's disease classification. *Çukurova Üniversitesi Mühendislik Fakültesi Dergisi*, 39(2):489–501, 2024.

[115] Liangming Pan, Alon Albalak, Xinyi Wang, and William Yang Wang. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning, 2023. URL https://arxiv.org/abs/2305.12295.

[116] Yikang Pan, Liangming Pan, Wenhu Chen, Preslav Nakov, Min-Yen Kan, and William Yang Wang. On the risk of misinformation pollution with large language models. *arXiv preprint arXiv:2305.13661*, 2023.

[117] Nicola Parenti, Roberta Manfredi, Maria Letizia Bacchi Reggiani, Diego Sangiorgi, and Tiziano Lenzi. Reliability and validity of an italian four-level emergency triage system. *Emergency Medicine Journal*, 27(7):495–498, 2010.

[118] Joonbum Park and Taeho Lim. Korean triage and acuity scale (ktas). *Journal of The Korean Society of Emergency Medicine*, 28(6):547–551, 2017.

[119] Jesse M. Pines, Joshua A. Hilton, Ellen J. Weber, Annechien J. Alkemade, Hasan Al Shabanah, Philip D. Anderson, Michael Bernhard, Alessio Bertini, André Gries, Santiago Ferrandiz, Vijaya Arun Kumar, Veli-Pekka Harjola, Barbara Hogan, Bo Madsen, Suzanne Mason, Gunnar Öhlén, Timothy Rainer, Niels Rathlev, Eric Revue, Drew Richardson, Mehdi Sattarian, and Michael J. Schull. International perspectives on emergency department crowding. *Academic Emergency Medicine*, 18(12):1358–1370, 2011. doi: https://doi.org/10.1111/j.1553-2712.2011.01235.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1553-2712.2011.01235.x.

[120] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. Pre-trained models for natural language processing: A survey. *Science China technological sciences*, 63(10):1872–1897, 2020.

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

[122] 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.

[123] Yoshihiko Raita, Tadahiro Goto, Mohammad Kamal Faridi, David FM Brown, Carlos A Camargo, and Kohei Hasegawa. Emergency department triage prediction of clinical outcomes using machine learning models. *Critical care*, 23:1–13, 2019.

[124] Adhitya Ramamurthi, Chandrakanth Are, and Anai N Kothari. From chatgpt to treatment: the future of ai and large language models in surgical oncology. *Indian journal of surgical oncology*, 14(3):537–539, 2023.

[125] Rajesh Ranjan, Shailja Gupta, and Surya Narayan Singh. A comprehensive survey of bias in llms: Current landscape and future directions. *arXiv preprint arXiv:2409.16430*, 2024.

[126] Raymond Reiter. A logic for default reasoning. *Artificial intelligence*, 13 (1-2):81–132, 1980.

[127] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Model-agnostic interpretability of machine learning. *arXiv preprint arXiv:1606.05386*, 2016.

[128] Pablo Rivas and Liang Zhao. Marketing with chatgpt: Navigating the ethical terrain of gpt-based chatbot technology. *AI*, 4(2):375–384, 2023.

[129] Lior Rokach and Oded Maimon. Decision trees. *Data mining and knowledge discovery handbook*, pages 165–192, 2005.

[130] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, 2019. URL `https://arxiv.org/abs/1811.10154`.

[131] Ji Ryu, Mun-Ki Min, Dae-Sup Lee, Seok Yeom, Seong-Hwa Lee, Il Jae Wang, Suck-Ju Cho, Seong Hwang, Jun-Ho Lee, and Yong-Hwan Kim. Changes in relative importance of the 5-level triage system, korean triage and acuity scale, for the disposition of emergency patients induced by forced reduction in its level number: a multi-center registry-based retrospective cohort study. *Journal of Korean Medical Science*, 34, 04 2019. doi: 10.3346/jkms.2019.34.e114.

[132] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*, 2024.

[133] Rocco Savage. User prompts vs. system prompts: What's the difference?, October 2024. URL `https://www.regie.ai/blog/user-prompts-vs-system-prompts`. Accessed: 2024-03-02.

[134] scikit-learn developers. *RobustScaler*, 2025. URL `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html`. Accessed: 2025-03-12.

[135] Yon Hee Seo, Kangbum Lee, and Kyeongmin Jang. Factors influencing the classification accuracy of triage nurses in emergency department: analysis of triage nurses' characteristics. *BMC nursing*, 23(1):1–6, 2024.

[136] Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, et al. Scrolls: Standardized comparison over long language sequences. *arXiv preprint arXiv:2201.03533*, 2022.

[137] Chantal Shaib, Millicent L. Li, Sebastian Joseph, Iain J. Marshall, Junyi Jessy Li, and Byron C. Wallace. Summarizing, simplifying, and synthesizing medical evidence using gpt-3 (with varying success), 2023. URL `https://arxiv.org/abs/2305.06299`.

[138] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

[139] Sonish Sivarajkumar and Yanshan Wang. Healthprompt: A zero-shot learning paradigm for clinical natural language processing, 2022. URL `https://arxiv.org/abs/2203.05061`.

[140] Carrie Stetler. Ai algorithms used in healthcare can perpetuate bias, 2024. URL `https://www.newark.rutgers.edu/news/ai-algorithms-used-healthcare-can-perpetuate-bias`.

[141] Krisada Suamchaiyaphum, Allison R Jones, and Adelais Markaki. Triage accuracy of emergency nurses: an evidence-based review. *Journal of Emergency Nursing*, 50(1):44–54, 2024.

[142] Lichao Sun, Yue Huang, Haoran Wang, Siyuan Wu, Qihui Zhang, Chujie Gao, Yixin Huang, Wenhan Lyu, Yixuan Zhang, Xiner Li, et al. Trustllm: Trustworthiness in large language models. *arXiv preprint arXiv:2401.05561*, 3, 2024.

[143] Hon Lon Tam, Siu Fung Chung, and Chi Kin Lou. A review of triage accuracy and future direction. *BMC Emergency Medicine*, 18:1–7, 2018.

[144] Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. Large language models in medicine. *Nature medicine*, 29(8):1930–1940, 2023.

[145] Maciej Tomczak and Ewa Tomczak. The need to report effect size estimates revisited. an overview of some recommended measures of effect size. 2014.

[146] Petter Törnberg. How to use llms for text analysis. *arXiv preprint arXiv:2307.13106*, 2023.

[147] Debbie A Travers, Anna E Waller, J Michael Bowling, Deborah Flowers, and Judith Tintinalli. Five-level triage system more effective than three-level in tertiary emergency department. *Journal of emergency nursing*, 28 (5):395–400, 2002.

[148] Ehsan Ullah, Anil Parwani, Mirza Mansoor Baig, and Rajendra Singh. Challenges and barriers of using large language models (llm) such as chatgpt for diagnostic medicine with a focus on digital pathology–a recent scoping review. *Diagnostic pathology*, 19(1):43, 2024.

[149] Andrea Valenzuela. Chain-of-thought prompting: Step-by-step reasoning with llms, 2024. URL `https://www.datacamp.com/tutorial/chain-of-thought-prompting`. Accessed: 2025-02-21.

[150] Ineke van der Wulp and Henk F van Stel. Calculating kappas from adjusted data improved the comparability of the reliability of triage systems: a comparative study. *Journal of clinical epidemiology*, 63(11):1256–1263, 2010.

[151] 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.

[152] Giulia Vilone and Luca Longo. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76: 89–106, 2021.

[153] Changyu Wang, Siru Liu, Hao Yang, Jiulin Guo, Yuxuan Wu, and Jialin Liu. Ethical considerations of using chatgpt in health care. *Journal of Medical Internet Research*, 25:e48009, 2023.

[154] Jiaqi Wang, Enze Shi, Sigang Yu, Zihao Wu, Chong Ma, Haixing Dai, Qiushi Yang, Yanqing Kang, Jinru Wu, Huawen Hu, et al. Prompt engineering for healthcare: Methodologies and applications. *arXiv preprint arXiv:2304.14670*, 2023.

[155] 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.

[156] Yeqi Wang, Weijian Huang, Cheng Li, Xiawu Zheng, Yusong Lin, and Shanshan Wang. Adaptive promptnet for auxiliary glioma diagnosis without contrast-enhanced mri, 2022. URL `https://arxiv.org/abs/2211.07966`.

[157] 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.

[158] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. 2023. URL `https://arxiv.org/abs/2201.11903`.

[159] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*, 2023.

[160] Wikipedia. Multimodal learning — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Multimodal%20learning&oldid=1253088850`, 2024. [Online; accessed 24-February-2025].

[161] Wikipedia. GPT-4o — Wikipedia, the free encyclopedia. http://en.wikipedia.org/w/index.php?title=GPT-4o&oldid=1276810101, 2025. [Online; accessed 24-February-2025].

[162] Wikipedia contributors. Prompt injection, 2025. URL https://en.wikipedia.org/wiki/Prompt_injection. Accessed: 2025-03-10.

[163] Christopher Y. K. Williams, Travis Zack, Brenda Y. Miao, Madhumita Sushil, Michelle Wang, Aaron E. Kornblith, and Atul J. Butte. Use of a Large Language Model to Assess Clinical Acuity of Adults in the Emergency Department. *JAMA Network Open*, 7(5):e248895–e248895, 05 2024. ISSN 2574-3805. doi: 10.1001/jamanetworkopen.2024.8895. URL https://doi.org/10.1001/jamanetworkopen.2024.8895.

[164] Brandon Wohlwend. Decision tree, random forest, and xgboost: An exploration into the heart of machine learning. Medium, 2023.

[165] Jiageng Wu, Xian Wu, and Jie Yang. Guiding clinical reasoning with large language models via knowledge seeds, 2024. URL https://arxiv.org/abs/2403.06609.

[166] Amit Yadav. Shap values vs feature importance. https://medium.com/biased-algorithms/shap-values-vs-feature-importance-ba6b91c16319, September 2024.

[167] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers, 2024. URL https://arxiv.org/abs/2309.03409.

[168] Rui Yang, Ting Fang Tan, Wei Lu, Arun James Thirunavukarasu, Daniel Shu Wei Ting, and Nan Liu. Large language models in health care: Development, applications, and challenges. *Health Care Science*, 2(4):255–263, 2023.

[169] Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony B Costa, Mona G Flores, et al. A large language model for electronic health records. *NPJ digital medicine*, 5(1):194, 2022.

[170] Jia-Yu Yao, Kun-Peng Ning, Zhen-Hui Liu, Mu-Nan Ning, and Li Yuan. Llm lies: Hallucinations are not bugs, but features as adversarial examples. *arXiv preprint arXiv:2310.01469*, 2023.

[171] Gokul Yenduri, Ramalingam M, Chemmalar Selvi G, Supriya Y, Gautam Srivastava, Praveen Kumar Reddy Maddikunta, Deepti Raj G, Rutvij H Jhaveri, Prabadevi B, Weizheng Wang, Athanasios V. Vasilakos, and Thippa Reddy Gadekallu. Generative pre-trained transformer: A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions, 2023. URL https://arxiv.org/abs/2305.10435.

[172] Vithya Yogarajan, Gillian Dobbie, and Te Taka Keegan. Debiasing large language models: research opportunities. *Journal of the Royal Society of New Zealand*, 55(2):372–395, 2025.

[173] Hyoungju Yun, Jinwook Choi, Jeong Ho Park, et al. Prediction of critical care outcome for adult patients presenting to emergency department using initial triage information: an xgboost algorithm analysis. *JMIR Medical Informatics*, 9(9):e30770, 2021.

[174] İlker Yıldız. Triage application with machine learning models, 2022. URL `https://www.kaggle.com/code/ilkeryildiz/triage-application-with-machine-learning-models/notebook`.

[175] Joany M. Zachariasse, Nienke Seiger, Pleunie P. M. Rood, Claudio F. Alves, Paulo Freitas, Frank J. Smit, Gert R. Roukema, and Henriëtte A. Moll. Validity of the manchester triage system in emergency care: A prospective observational study. *PLOS ONE*, 12(2):1–14, 02 2017. doi: 10.1371/journal.pone.0170811. URL `https://doi.org/10.1371/journal.pone.0170811`.

[176] Joany M Zachariasse, Vera van der Hagen, Nienke Seiger, Kevin Mackway-Jones, Mirjam van Veen, and Henriette A Moll. Performance of triage systems in emergency care: a systematic review and meta-analysis. *BMJ open*, 9(5):e026471, 2019.

[177] Hugh Zhang and David C. Parkes. Chain-of-thought reasoning is a policy improvement operator, 2023. URL `https://arxiv.org/abs/2309.08589`.

[178] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*, 2022.

[179] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2024. URL `https://arxiv.org/abs/2303.18223`.

[180] Ce Zhou, Qian Li, Chen Li, Jun Yu, Yixin Liu, Guangjing Wang, Kai Zhang, Cheng Ji, Qiben Yan, Lifang He, Hao Peng, Jianxin Li, Jia Wu, Ziwei Liu, Pengtao Xie, Caiming Xiong, Jian Pei, Philip S. Yu, and Lichao Sun. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt, 2023. URL `https://arxiv.org/abs/2302.09419`.

[181] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. Least-to-most prompting enables complex reasoning in large language models, 2023. URL `https://arxiv.org/abs/2205.10625`.

[182] İlker Yıldız. Emergency service - triage application, 2019. URL https://www.kaggle.com/datasets/ilkeryildiz/emergency-service-triage-application.