

Nearest-Neighbor Case Retrieval for Emergency Department Urgency Prediction with a Large Language Model

Ismail Bouajaja¹, Omar Bouattane², Oussama Barakat³, Abdelaziz Daai⁴

IESI Laboratory, ENSET, Hassan II University of Casablanca, Mohammedia, Morocco^{1,2}

SINERGIES Laboratory, Marie & Louis Pasteur University, Besançon, France^{1,3}

2IACS Laboratory, ENSET, Hassan II University of Casablanca, Mohammedia, Morocco⁴

Abstract—This study evaluates whether prior emergency department visits improve a large language model's (LLM) prediction of recorded urgency. Four methods were compared: an always-urgent baseline, conventional k-nearest-neighbor (kNN) voting, an LLM without retrieved cases, and the same LLM with 30 kNN-retrieved labeled cases. The 2022 National Hospital Ambulatory Medical Care Survey yielded 10,207 eligible visits: 7,167 for training, 1,522 for validation, and 1,518 for testing. The final case library combined the 8,689 training and validation visits. On the commonly held-out test set, the kNN-search LLM achieved 74.70% accuracy and 66.86% balanced accuracy, compared with 68.97% and 65.68% for the no-search LLM and 62.52% and 64.30% for conventional kNN. Relative to no-search prompting, retrieved cases increased urgent sensitivity from 73.87% to 86.37% but reduced nonurgent specificity from 57.49% to 47.36%. A paired 95% interval excluded zero for the accuracy difference but included zero for balanced accuracy. API-reported model-call charges were USD 0.0786 without cases and USD 0.8577 with cases. The results demonstrate a change in the balance of errors, not established clinical utility or uniformly better discrimination.

Keywords—Emergency department; urgency prediction; large language models; nearest neighbors; case retrieval; tabular classification

I. INTRODUCTION

Emergency department prediction studies address different targets. Recorded triage categories describe assessed priority for care, whereas admission and critical-care outcomes occur later and are not interchangeable targets. The National Hospital Ambulatory Medical Care Survey (NHAMCS) provides public-use emergency department records [1] used to study both recorded urgency [2] and later clinical outcomes [3].

The target in this study is the recorded binary urgency label (urgent or nonurgent), not an independent medical judgment. The analysis examines whether a pretrained large language model (LLM) can use labeled prior visits to predict that label. Each case is a recorded visit, and retrieval selects labeled cases from a stored library.

Four methods are compared. The frequency-only baseline always predicts the most frequent fitting label, urgent. Conventional k-nearest neighbors (kNN) retrieves nearby visits and applies a fixed voting threshold. The no-search LLM receives the current visit and field definitions but no examples.

The kNN-search LLM receives the current visit and the same ordered labeled cases retrieved by conventional kNN, then produces its own label rather than applying the kNN vote.

Table I distinguishes case selection from the final decision. In the kNN-search procedure, retrieval occurs before the LLM call; the model cannot request additional cases. Neither LLM procedure updates model weights. The term kNN-search LLM therefore denotes fixed external retrieval followed by an LLM decision, not model-controlled search.

The primary hypothesis examined is that providing thirty fixed nearest-neighbor cases increases the LLM's accuracy and balanced accuracy relative to presenting the current visit alone on the same held-out sample. A secondary comparison contrasts LLM decisions with conventional kNN voting on identical ordered neighbors. The tests concern complete prediction procedures: they do not isolate the causal effect of neighbor similarity, input representation, or pretrained knowledge. Class-specific errors and the added model-call charges are also reported.

II. RELATED WORK

A. Prediction from Emergency-Department Records

Zhang et al. [2] studied a related binary urgency target using 8,716 adult visits from another NHAMCS year. They grouped levels 1–3 versus 4–5 and tested logistic regression, random forests, gradient boosting, and extreme gradient boosting with structured and textual inputs. Gradient boosting with combined inputs achieved an area under the receiver operating characteristic curve of 0.789 and an accuracy of 0.726 in fivefold cross-validation. Differences in cohort, inputs, and evaluation prevent a direct performance ranking.

TABLE I. SEARCH AND DECISION IN THE FOUR METHODS

Method	Search	Decision
Frequency-only baseline	None	Always urgent
Conventional kNN	Predefined nearest-neighbor retrieval	Equal votes and threshold
No-search LLM	None	LLM label from current visit
kNN-search LLM	Same fixed neighbors as kNN	LLM label from visit and examples

Raita et al. [3] used NHAMCS 2007–2015 to predict critical-care and hospitalization outcomes from triage information. Those later outcomes differ from the recorded binary urgency label studied here, so their scores are not direct benchmarks.

B. Tabular Prediction and Example Selection

Hegselmann et al. [4] introduced TabLLM, which serializes table rows as text for language-model classification. They compared input formats and evaluated tasks with few or no labeled examples; their few-shot configuration used fine-tuning. In contrast, both LLM methods in the present study retain fixed model weights, and only one receives retrieved examples in the prompt.

Liu et al. [5] investigated how selecting in-context examples affects GPT-3. They retrieved examples with similar textual representations and reported improvements over random selection on their evaluated language tasks. These findings motivate case selection but do not establish an appropriate similarity measure or case count for emergency visits.

Wu and Hou [6] proposed retrieval-based tabular prediction with an LLM and no task-specific model training. Their method selects target-dataset examples for a prediction prompt. The present comparison additionally gives a conventional voting rule and an LLM exactly the same retrieved case list.

C. Language-Model Triage and Clinical Recommendations

Lu et al. [7] introduced TriageAgent, combining LLM agents with retrieval from an Emergency Severity Index handbook. The present experiment instead uses one LLM with either no examples or a fixed set of labeled visits, without handbook retrieval or agent cooperation.

Lee et al. [8] evaluated no-example prompting, example prompting, nearest-neighbor retrieval, and fine-tuning for emergency-department triage using several clinical datasets. Their nearest-case method selects candidates from biomedical complaint-text embeddings and refines them using normalized vital signs; missingness and demographic biases are also examined. This work precedes the present retrieved-case comparison, although its five-level targets and retrieval method differ from the binary NHAMCS setting.

Williams et al. [9] evaluated GPT-3.5-turbo and GPT-4-turbo recommendations based on emergency-department notes, including admission, imaging, and antibiotic decisions. They observed sensitivity–specificity trade-offs that varied by task and prompt. Those recommendation targets are not equivalent to recorded triage urgency, but the trade-offs inform interpretation of the present results.

Previous work has established tabular LLM prediction, example retrieval, and LLM-assisted triage. The narrower contribution here is a paired comparison of two decision procedures on the same ordered neighbors, alongside a comparison of one LLM with and without supplied cases. Random examples, alternative retrieval measures, and other LLMs were not evaluated. Consequently, the study does not identify the benefit of neighbor similarity alone or establish an optimal retrieval design.

III. DATA AND STUDY PROTOCOL

A. Dataset and Target

The dataset is the 2022 NHAMCS emergency-department public-use release from the National Center for Health Statistics (NCHS) [1], containing 16,025 visit records and 913 fields. Each record represents a visit, not necessarily a unique patient. Eligibility requires a usable recorded urgency category, but not complete predictor or diagnosis fields; all ages are included.

IMMEDR records five urgency categories: immediate (1), emergent (2), urgent (3), semi-urgent (4), and nonurgent (5) [1, p. 36]. Categories 1–3 were grouped as urgent (1), and 4–5 as nonurgent (0), following Zhang et al. [2]. The 5,818 records without a usable category were excluded, leaving 10,207 visits (7,074 urgent and 3,133 nonurgent). The mechanism behind missing urgency labels was not assessed, and the retained population cannot be assumed representative of the excluded visits [10]. These categories are recorded labels, not independently verified clinical judgments.

All descriptive counts and scores are unweighted; NHAMCS survey weights, needed for national estimates, were not applied [1]. Results therefore characterize the eligible released visits rather than all US emergency visits. Admission, mortality, and independently assessed urgency were not evaluated.

B. Selected Inputs

Table II lists thirteen inputs covering age, recorded sex, ambulance arrival, the first complaint, initial measurements, pain, and recent attendance [1]. These were selected to approximate information available at initial assessment. Diagnoses, later investigations, treatments, disposition, identifiers, and the urgency label were excluded from predictors. Omitted complaints and comorbidities may nevertheless carry useful information.

Pre-triage availability of every selected input cannot be confirmed for every record. Documented missing and nonmeasurement codes were interpreted before processing; usable values already imputed by NCHS were retained. Ages of 93 or more, including the released 94-or-older category, were encoded as 93 and displayed as “93 years or older”. Temperature was divided by ten and retained in Fahrenheit. Because the audited blood-pressure definitions did not state units, the prompt described those units as unspecified.

C. Development and Held-out Data

Eligible visits were partitioned into approximately 70% training, 15% validation, and 15% held-out testing using random seed 42, stratified by urgency and primary-diagnosis availability. Diagnosis availability was used only for stratification. Training data fitted the initial representation, validation data selected kNN settings, and test data supported the final comparison (Table III).

The final case library combines training and validation rather than forming another independent partition. All methods use the same 1,518 held-out visits, including 1,064 urgent visits (70.09%). Class proportions differ slightly across partitions.

TABLE II. SOURCE INPUTS

Field	Meaning and display
AGE	Age in years; model cap displayed as 93 years or older
AGEDAYS	Age in days below one year; otherwise not applicable
SEX	Recorded sex; code and meaning
ARREMS	Arrival by ambulance; code and meaning
RFV1	First complaint or reason for attending; not a diagnosis
TEMPF	Initial temperature, degrees Fahrenheit (recorded scale)
PULSE	Initial heart rate, beats per minute
RESPR	Initial breathing rate, breaths per minute
BPSYS	Initial systolic blood pressure; unit not specified
BPDIAS	Initial diastolic blood pressure; unit not specified
POPCT	Initial oxygen saturation, percent
PAINSCALE	Pain intensity on the recorded zero-to-ten scale
SEEN72	Prior visit to the same department within 72 hours

TABLE III. DATA PARTITIONS

Population	Visits	Urgent	Nonurgent
Training	7,167	4,959	2,208
Validation	1,522	1,051	471
Final case library	8,689	6,010	2,679
Held-out test	1,518	1,064	454

KNN settings were selected against the validation partition using training cases only. Preprocessing was then refitted on the combined 8,689 training and validation visits for final retrieval, without changing the chosen k or decision threshold. The no-search LLM received no cases. Each test visit used a separate conversation, and test labels were reserved for evaluation.

IV. PREDICTION METHODS

A. Frequency-only Baseline

The frequency-only baseline predicts urgent for every visit, reflecting the majority label in the final fitting set (6,010 of 8,689 visits). Its accuracy equals the urgent proportion in the evaluation population. This classifier has 100% urgent sensitivity, 0% nonurgent specificity, and 50% balanced accuracy; its apparent accuracy does not reflect discrimination.

B. Conventional Nearest Neighbors

Conventional kNN uses Euclidean distance and uniform voting [11], implemented with NumPy. The nine numerical inputs are AGE, AGEDAYS, TEMPF, PULSE, RESPR, BPSYS, BPDIAS, POPCT, and PAINSCALE. Missing numerical values are replaced with fitting-set medians, then standardized using post-filling means and population standard deviations from fitting data. A zero standard deviation uses a divisor of one.

Nine binary indicators distinguish median-filled numerical values from usable observations. AGEDAYS has a further indicator for nonapplicability. SEX, ARREMS, RFV1, and SEEN72 are one-hot encoded, with reserved indicators for

missing/unusable values and previously unseen valid categories. Table IV gives the encoded dimensions. These columns do not represent additional source variables.

TABLE IV. ENCODED INPUT DIMENSIONS

Component	Training-only	Final library
Numerical values	9	9
Filling indicators	9	9
AGEDAYS not applicable	1	1
SEX	4	4
ARREMS	4	4
RFV1	430	449
SEEN72	4	4
Total	461	480

RFV1 contributes 428 observed categories during training and 447 after refitting, plus two reserved indicator columns. Encoded vectors contain standardized values and binary indicators. Euclidean distances are calculated over all columns; the k closest records are selected, breaking exact ties by their original 2022 row numbers. No distance cutoff is applied. If $a(x)$ is the number of urgent labels among the k neighbors, their urgent proportion is:

$$s(x) = \frac{a(x)}{k} \quad (1)$$

The predicted label is urgent when $s(x)$ is at least threshold t . Validation considered $k = 5, 15$, and 30 and thresholds from 0.1 to 0.9 in increments of 0.1 (27 combinations), maximizing balanced accuracy. Exact ties favored smaller k , then a threshold closer to 0.5 , then the smaller threshold.

C. Common Language-model Input and Output

The evaluated model was GPT-6 Luna through the OpenRouter application programming interface (API), requested as openai/gpt-6-luna with recorded catalog identifier openai/gpt-6-luna-20260922. Requests were restricted to the standard OpenAI provider without fallback; reasoning effort was set to none, the output limit was 128 tokens, and responses were not streamed. Temperature and top-p were not specified in the recorded requests. Therefore, deterministic or repeatable sampling is not assumed. Model weights were unchanged.

The instruction defined urgent as 1 and nonurgent as 0, with field meanings, available units, and category labels. Unusable values were explicitly identified rather than numerically imputed for the LLM. The prompt omitted NHAMCS, IMMEDR, the source year, and the original five-level mapping; this does not exclude possible exposure to public data during pretraining.

Each accepted response was a structured JSON object containing only the integer field "urgency" (0 or 1), enforced through the API and checked locally. Labels were used without a post hoc threshold or requested explanation. Prompts were fixed without LLM validation tuning; Appendix A reproduces their instructions.

D. Language Model Without Retrieved Cases

The no-search LLM receives only the thirteen readable input fields and fixed instructions. It receives no labeled examples, class proportions, kNN scores, or search tools. Although no task-specific cases are supplied, the model retains knowledge learned during pretraining.

E. Language Model with Fixed Nearest-Neighbor Cases

The kNN-search LLM receives exactly the thirty ordered cases used for conventional kNN. Each labeled example includes all thirteen readable inputs and its urgency. Examples appear from nearest to farthest, followed by the unlabeled visit to predict. Source identifiers and numerical distances are omitted.

The prompt does not disclose the kNN vote, score, threshold, or prediction, and it does not instruct the LLM to follow the majority class. The neighbor count ($k = 30$) was selected through conventional kNN validation, not optimized separately for the LLM. Thus, retrieval is shared, but decision rules, representations, and pretrained knowledge differ.

V. EVALUATION AND EXECUTION

A. Shared Test Comparison

All four procedures were evaluated on the same 1,518 held-out visits, without adjusting kNN settings or LLM prompts to test results. The primary comparison is kNN-search versus no-search prompting with the same LLM; the secondary comparison is kNN-search LLM versus conventional kNN voting.

Urgent is the positive class. Table V defines accuracy, urgent precision and sensitivity, nonurgent specificity, urgent F1, and balanced accuracy using true-positive, false-negative, false-positive, and true-negative counts. Undefined measures are reported as not applicable.

TABLE V. BINARY PERFORMANCE MEASURES

Measure	Calculation; urgent is positive
Accuracy	(TP + TN) divided by completed visits
Urgent precision	TP divided by (TP + FP)
Urgent recall / sensitivity	TP divided by (TP + FN)
Nonurgent specificity	TN divided by (TN + FP)
Urgent F1	2 TP divided by (2 TP + FP + FN)
Balanced accuracy	Mean of sensitivity and specificity

Each valid predicted label was compared with the recorded reference label. All four procedures predicted all 1,518 assigned visits, so completed and assigned evaluation denominators coincide.

B. Calls, Retries and Resources

Visits were processed independently with up to 32 concurrent LLM requests. Failed calls could be retried with identical request content; accepted labels were not regenerated. Completed outputs were retained when execution resumed.

Attempt-level token usage, API charges, and elapsed times were logged when available. Request latency measures one

call, whereas runner wall time measures the active request-processing program. Data loading, retrieval, and prompt construction were not timed end to end for the final LLM runs, and remote inference memory was not measured. The timing figures therefore cannot support end-to-end performance comparisons. Reporting follows the transparency aims of Gallifant et al. [12], without claiming full guideline compliance.

A descriptive uncertainty analysis used 10,000 paired bootstrap draws stratified by the recorded label, preserving 1,064 urgent and 454 nonurgent visits [13]. Each draw used the same sampled indices for all methods (seed 42). The 2.5th and 97.5th percentiles of differences between kNN-search and each comparator formed descriptive 95% intervals. This post hoc analysis did not change predictions or settings and does not account for dependence among visits within hospitals.

VI. RESULTS

A. Completed Nearest-Neighbor Validation

All 27 settings were evaluated on 1,522 validation visits against 7,167 training cases. The highest balanced accuracy occurred at $k = 30$ and $t = 0.7$, which classifies a visit as urgent when at least 21 neighbors are urgent. Table VI reports the best threshold for each k ; Fig. 1 shows the full threshold comparison.

TABLE VI. BEST VALIDATION SETTING FOR EACH NEIGHBOR COUNT

k	Threshold	Balanced accuracy
5	0.7	66.4195%
15	0.7	68.2828%
30	0.7	68.8682%

For $k = 5$, thresholds 0.7 and 0.8 gave identical predictions, so the stated tie rule retained 0.7. These are validation, not test, results.

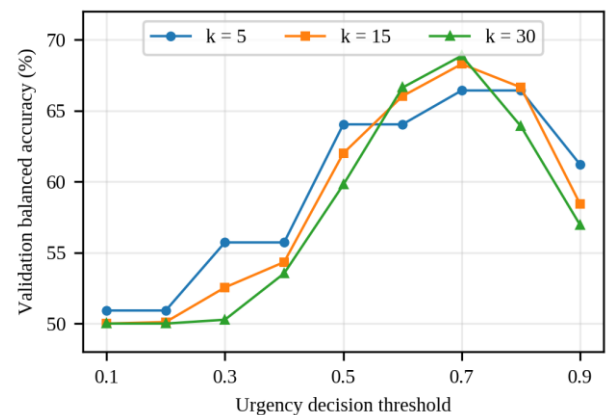


Fig. 1. Validation balanced accuracy across decision thresholds. Each line uses a different neighbor count.

At the selected setting, validation yielded 67.2799% accuracy, 84.2627% precision, 64.7003% sensitivity, 73.0361% specificity, 73.1970% F1, and 68.8682% balanced accuracy (680 TP, 371 FN, 127 FP, 344 TN).

Refitting on all 8,689 training and validation visits produced 480 encoded columns. The selected k and t remained fixed for the test set.

B. Recorded Development Execution

The local kNN execution used Python 3.12.11 and NumPy 2.5.3 in Linux Docker on an Intel Core i7-7700HQ. The recorded validation search and scoring times were 9.4980 and 0.0212 seconds, respectively; final refitting took 0.0718 seconds.

An uncached validation pass required 9.8845 seconds for 1,522 visits. Local memory and stored-model size were measured only during development and are not comparable to remote LLM inference costs.

C. Held-out Prediction Results

All four methods predicted all 1,518 test visits (Table VII). The kNN-search LLM had the highest observed accuracy, F1, and balanced accuracy, but not the highest precision, sensitivity, or specificity. Performance therefore depends on the error type of interest.

The always-urgent classifier achieved 70.0922% accuracy, matching the urgent-label prevalence, while correctly identifying no nonurgent visits. Conventional kNN achieved 64.2954% test balanced accuracy, below its 68.8682% validation value; its threshold was not revised.

Table VIII reports paired percentage-point (pp) differences between kNN-search and each comparator. Accuracy exceeded that of no-search LLM by 5.73 pp and conventional kNN by 12.19 pp; the balanced-accuracy differences were 1.18 and 2.57 pp. Both accuracy intervals exclude zero, whereas both balanced-accuracy intervals include zero.

TABLE VII. HELD-OUT METRICS ON ALL 1,518 VISITS

Metric (%)	Always urgent	kNN	No-search LLM	kNN-search LLM
Accuracy	70.09	62.52	68.97	74.70
Urgent precision	70.09	81.77	80.29	79.36
Urgent sensitivity	100.00	59.87	73.87	86.37
Nonurgent specificity	0.00	68.72	57.49	47.36
Urgent F1	82.42	69.13	76.95	82.72
Balanced accuracy	50.00	64.30	65.68	66.86

TABLE VIII. PAIRED SCORE DIFFERENCES FOR THE KNN-SEARCH LLM

Comparator	Metric	Change (pp)	95% interval (pp)
kNN	Accuracy	+12.19	+9.62 to +14.76
kNN	Balanced accuracy	+2.57	-0.24 to +5.44
No-search LLM	Accuracy	+5.73	+3.43 to +8.17
No-search LLM	Balanced accuracy	+1.18	-1.64 to +4.09

Table IX gives the exact confusion counts for all four methods. Relative to the no-search LLM, the kNN-search LLM corrected 225 errors but changed 138 correct predictions to errors (a net gain of 87). It correctly predicted 133 additional urgent visits but 46 fewer nonurgent visits, increasing sensitivity by 12.50 pp and decreasing specificity by 10.13 pp.

TABLE IX. HELD-OUT CONFUSION COUNTS

Method	TP	FN	FP	TN
Frequency-only baseline	1064	0	454	0
Conventional kNN	637	427	142	312
No-search LLM	786	278	193	261
kNN-search LLM	919	145	239	215

D. Completion, Cost and Execution Time

Table X reports API usage and charges. The no-search LLM required one retry following a timeout with unreported usage, while the kNN-search LLM required none. Both returned 1,518 valid predictions. Reported charges exclude preprocessing, retrieval, and earlier technical checks.

Token and cost totals represent the 1,518 reported no-search calls and all 1,518 kNN-search calls. The unreported timed-out attempt could have incurred an additional charge; its amount is unknown.

The frequency-only and conventional kNN commands took 0.3448 and 12.9060 seconds, respectively. Table XI reports LLM runner and request timings, whose boundaries exclude some retrieval and formatting steps. No matched end-to-end speed comparison is available.

TABLE X. REPORTED LLM USAGE AND CHARGES

Quantity	No-search LLM	kNN-search LLM
Model-call attempts	1,519	1,518
Failed calls/retries	1 / 1	0 / 0
Completed visits	1,518	1,518
Input (prompt) tokens	672,538	6,775,866
Output tokens	22,782	22,770
Reported cached tokens	0	4,520
Reported reasoning tokens	0	0
API-reported cost (USD)	0.0786448	0.8577346

TABLE XI. FINAL LLM TIMINGS IN SECONDS

Quantity	No-search LLM	kNN-search LLM
Request-runner wall time	403.7028	286.8920
Mean request latency	3.6476	4.0342
Median request latency	1.2587	1.4383
95th-percentile request	4.0636	3.9985
Mean visit elapsed	3.6896	4.0733
Median visit elapsed	1.2937	1.4717
95th-percentile visit	4.0913	4.0266

Overlapping requests make runner wall time different from the sum of visit latencies. The no-search timeout also lengthened runner time. The recorded timings do not establish faster end-to-end prediction for the kNN-search model.

VII. DISCUSSION

A. What Changed When Examples Were Supplied

Conventional kNN and the kNN-search LLM used the same ordered 30 neighbors for each test visit (45,540 case

positions). Compared with kNN voting, the LLM correctly predicted 282 more urgent visits but 97 fewer nonurgent visits. This is a shift in the error trade-off, not a uniform improvement.

Compared with the no-search LLM, supplying cases increased accuracy and urgent sensitivity but reduced urgent precision and nonurgent specificity. The balanced-accuracy improvement was small, and its descriptive interval included zero; the same applied to the comparison with conventional kNN. The strongest observed difference concerns overall accuracy, not established gains in discrimination across both classes.

Zhang et al. [2] reported 72.60% accuracy for gradient boosting with combined inputs. The kNN-search LLM's 74.70% does not establish superiority because the cohorts, age eligibility, inputs, and validation procedures differ. Only the four methods evaluated on the same 2022 held-out visits permit a direct within-study comparison.

B. Limits of the Study

The retrospective analysis uses one survey year, one split, and no hospital-level or patient-level holdout. Repeated visits may belong to the same patient. The 5,818 records without usable urgency labels were excluded, and differences between included and excluded visits were not measured; selection bias therefore remains possible [10]. Survey nonresponse and unweighted analysis also restrict generalizability [1]. Bootstrap intervals do not adjust for hospital-level dependence.

The binary target combines five recorded categories, whereas the prompt states only "urgent" and "nonurgent"; the model's interpretation may therefore differ from the survey grouping. Pre-label availability of all inputs could not be established. NCHS reports sex imputation using grouped diagnosis and triage among the matching variables [1, p. 16]. Its effect on this evaluation was not measured, leaving possible target-related information in source preprocessing.

The LLM reads complaint descriptions, whereas kNN uses one-hot complaint codes. These representation differences, along with different prior knowledge and decision rules, prevent attribution of the performance gap to the decision mechanism alone. Operating points were not matched. Random-case controls, an LLM-specific neighbor count, other model versions, and repeated completions were not evaluated, so the specific value of nearest-case selection and output stability remain unresolved.

Agreement with recorded urgency labels is not evidence of clinical benefit. Outcomes, clinician agreement, and treatment decisions were not evaluated. The study does not validate autonomous clinical triage.

VIII. CONCLUSION

Four procedures were compared on the same 1,518 held-out visits. The kNN-search LLM used the thirty neighbors selected by conventional kNN and achieved the highest observed accuracy (74.70%) and balanced accuracy (66.86%) among the evaluated methods.

Relative to no-search prompting, retrieved cases reduced missed urgent labels but increased false urgent labels. The accuracy difference was clearer than the small and uncertain balanced-accuracy difference. API-reported charges for all kNN-search LLM calls totalled USD 0.8577. Validation across hospitals, model versions, and repeated runs is needed before assessing clinical applicability.

DATA AND CODE AVAILABILITY

The study used public-use NHAMCS records and documentation [1], without patient contact or re-identification. Code and remaining experiment artefacts have not been publicly released.

DECLARATION ON GENERATIVE AI

ChatGPT was used to assist with manuscript drafting, language revision, and formatting.

Based on implementation plans approved by the authors, OpenCode with GPT-5.6 Sol Xhigh assisted with implementing and running the experiments. GPT-6 Luna was used as the prediction model, as described in Section IV.

The authors critically reviewed and edited the AI-assisted material, checked the implementation and reported results against the experimental records, and assessed the scientific arguments and conclusions. The authors take full responsibility for the accuracy, originality, and integrity of the work. No AI tool is listed as an author.

APPENDIX A: EXACT PROMPT AND EVIDENCE ORDER

The following text reproduces the no-search LLM system instruction. The kNN-search LLM used the same text with the additional block below inserted before the output-format paragraph. Line wrapping is for display.

Predict the urgency of this emergency-department visit.

The target:

- 1 means urgent.
- 0 means nonurgent.

Input fields:

- AGE: patient age in years.
- AGEDAYS: age in days for patients younger than one year.
- SEX: recorded patient sex.
- ARREMS: whether the patient arrived by ambulance.
- RFV1: the first recorded complaint or reason for the visit.
- TEMPF: initial body temperature in degrees Fahrenheit.
- PULSE: initial heart rate, in beats per minute.
- RESPR: initial breathing rate, in breaths per minute.
- BPSYS: initial systolic blood pressure.
- BPDIAS: initial diastolic blood pressure.

- POPCT: initial oxygen saturation from pulse oximetry, in per cent.
- PAINSCALE: reported pain intensity on a 0-10 scale.
- SEEN72: whether the patient visited the same emergency department within the previous 72 hours.

The next message gives the values, with meanings, units or scales where available. "Unavailable" means there is no usable value. "Not applicable" means the field does not apply to this visit. Do not invent patient information.

Return only a JSON object with one key, "urgency". Its value must be the integer 0 or 1. Do not include an explanation or other fields.

Additional instruction for the kNN-search LLM:

You also receive 30 example visits, ordered from nearest to farthest.

Each example includes its known urgency:

- 1 means urgent.
- 0 means nonurgent.

The example visits come first. The visit to assess is at the end. Use the visit to assess and these examples to predict its urgency.

The no-search LLM user message begins with "Visit to assess:" and the thirteen readable field lines. The kNN-search LLM user message begins with "Example visits, ordered from nearest to farthest:", followed by thirty numbered cases. Each has thirteen fields and "Urgency: 1 - urgent" or "Urgency: 0 - nonurgent". The current visit comes last with no label. Both messages end with that visit's SEEN72 line.

REFERENCES

- [1] National Center for Health Statistics, "2022 NHAMCS micro-data file documentation: Emergency department," Centers for Disease Control and Prevention. [Online]. Available: https://ftp.cdc.gov/pub/Health_Statistics/NCHS/Dataset_Documentation/NHAMCS/doc22-ed-508.pdf. Accessed: Sep. 25, 2026.
- [2] X. Zhang, Y. Wang, Y. Jiang, C. B. Pacella, and W. Zhang, "Integrating structured and unstructured data for predicting emergency severity: an association and predictive study using transformer-based natural language processing models," *BMC Medical Informatics and Decision Making*, vol. 24, Art. no. 372, 2024, doi: 10.1186/s12911-024-02793-9.
- [3] Y. Raita, T. Goto, M. K. Faridi, D. F. M. Brown, C. A. Camargo Jr., and K. Hasegawa, "Emergency department triage prediction of clinical outcomes using machine learning models," *Critical Care*, vol. 23, Art. no. 64, 2019, doi: 10.1186/s13054-019-2351-7.
- [4] S. Heggelmann, A. Buendia, H. Lang, M. Agrawal, X. Jiang, and D. Sontag, "TabLLM: few-shot classification of tabular data with large language models," in *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research*, vol. 206, 2023, pp. 5549–5581. [Online]. Available: <https://proceedings.mlr.press/v206/heggelmann23a.html>.
- [5] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, and W. Chen, "What makes good in-context examples for GPT-3?," in *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, Dublin, Ireland and Online: Association for Computational Linguistics, 2022, pp. 100–114, doi: 10.18653/v1/2022.deelio-1.10.
- [6] J. Wu and M. Hou, "An efficient retrieval-based method for tabular prediction with LLM," in *Proceedings of the 31st International Conference on Computational Linguistics*, Abu Dhabi, UAE: Association for Computational Linguistics, 2025, pp. 9917–9925. [Online]. Available: <https://aclanthology.org/2025.coling-main.663/>.
- [7] M. Lu, B. Ho, D. Ren, and X. Wang, "TriageAgent: towards better multi-agents collaborations for large language model-based clinical triage," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024, pp. 5747–5764, doi: 10.18653/v1/2024.findings-emnlp.329.
- [8] J. Lee, T. Shang, J. Y. Baik, D. Duong-Tran, S. Yang, L. Li, and L. Shen, "From promising capability to pervasive bias: assessing large language models for emergency department triage," *arXiv preprint arXiv:2504.16273v2*, 2025. [Online]. Available: <https://arxiv.org/abs/2504.16273v2>.
- [9] C. Y. K. Williams, B. Y. Miao, A. E. Kornblith, and A. J. Butte, "Evaluating the use of large language models to provide clinical recommendations in the Emergency Department," *Nature Communications*, vol. 15, Art. no. 8236, 2024, doi: 10.1038/s41467-024-52415-1.
- [10] D. B. Rubin, "Inference and missing data," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976, doi: 10.1093/biomet/63.3.581.
- [11] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967, doi: 10.1109/TIT.1967.1053964.
- [12] J. Gallifant et al., "The TRIPOD-LLM reporting guideline for studies using large language models," *Nature Medicine*, vol. 31, no. 1, pp. 60–69, 2025, doi: 10.1038/s41591-024-03425-5.
- [13] B. Efron, "Bootstrap methods: another look at the jackknife," in *Breakthroughs in Statistics*, S. Kotz and N. L. Johnson, Eds. New York, NY: Springer, 1992, pp. 569–593, doi: 10.1007/978-1-4612-4380-9_41.