Skepthical review: *Evaluating Attention-Based Learning of Patient Diagnosis Representations with Present On Admission Status for In-Hospital Mortality and Prolonged Length of Stay Prediction*

Summary

The paper studies prediction of in-hospital mortality and prolonged length of stay (PLOS) from the 2018 Texas Hospital Inpatient Discharge PUDF using administrative variables and diagnosis codes augmented with Present On Admission (POA) indicators (Sec. 1, Sec. 2.1–2.3). Diagnoses are truncated to 3-character ICD-10-CM categories and paired with six processed POA levels, including a dedicated missing/invalid category (POA_M_MISSING) (Sec. 2.3.2). A Transformer encoder is trained on a 1% random subsample ($\mathbf{31,!102}$ of $\sim \mathbf{3.1M}$ records) using in-hospital mortality as a proxy objective to produce a $\mathbf{160}$-dimensional patient embedding; this embedding is concatenated with engineered admission-time features (including geography encodings and a count of POA=$'Y'$ diagnoses) and used as input to downstream Logistic Regression and XGBoost models for mortality and PLOS prediction (Sec. 2.4–2.6). Baselines include (i) models using only non-diagnostic admission features and (ii) models augmenting these with sparse multi-hot encodings of the top-N principal and POA=$'Y'$ diagnoses (Sec. 2.6). On the held-out test split of the 1% subsample, the learned attention-based embedding does not outperform the simple diagnosis encodings—especially for mortality, where the attention-based downstream models underperform even the non-diagnosis baseline—while for PLOS the attention embedding yields only modest improvements over non-diagnosis features and remains below the sparse diagnosis baseline (Sec. 3.4). SHAP analysis highlights the dominance of engineered variables (notably NUM_POA_Y_DIAGNOSES, age, admission type/source, and geography encodings), with individual embedding dimensions contributing relatively little; attention-weight visualizations were reportedly not interpretable (Sec. 2.8, Sec. 3.5). The paper's main contribution is a careful negative result under compute/data constraints, suggesting that, in this specific experimental regime, POA-aware feature engineering can be difficult to beat with a constrained Transformer pipeline (Sec. 3.4, Sec. 4).

Strengths

- Important and practically motivated prediction problems (mortality, PLOS) with appropriate attention to class imbalance (Sec. 1, Sec. 2.2, Sec. 2.7).
- Sensible and clearly motivated handling of POA, including treating missing/invalid POA as an informative category (POA_M_MISSING), which matches real administrative data quality issues (Sec. 2.3.2).
- Strong, realistic baselines—especially Baseline 2 (top-N principal and POA=$'Y'$ diagnoses)—that make the negative result informative rather than vacuous (Sec. 2.6, Sec. 3.4).

- Use of PR-AUC as a primary metric for imbalanced mortality and reporting of AUC-ROC/AUC-PR for PLOS (Sec. 2.7, Sec. 3.4).
- Interpretability effort via SHAP that surfaces clinically/plausibly meaningful predictors (e.g., POA=$'Y'$ burden, age, admission source/type) and transparently shows limited marginal value from embedding dimensions (Sec. 2.8, Sec. 3.5).
- Overall narrative is candid about limitations (subsampling/compute) and does not over-claim improvements from deep learning (Sec. 3.4, Sec. 4).

Major issues

1. **Temporal availability / leakage risk: the manuscript repeatedly frames features as "known at admission," but the diagnoses used come from discharge abstracts and are typically coded over the entire stay. Even with POA indicators, the *presence and count* of recorded diagnoses (including secondary diagnoses up to 24) can reflect post-admission events, complications, documentation intensity, and length-of-stay itself. In particular, NUM_POA_Y_DIAGNOSES and inclusion of all secondary diagnoses may encode severity/coding intensity that is not truly available at admission, and may inflate performance or change the interpretation from early prediction to retrospective risk adjustment (Sec. 1, Sec. 2.3.2, Sec. 2.6, Sec. 4).**

   Recommendation: Clarify in Sec. 1 and Sec. 2.2 what "available at admission" means in the context of PUDF discharge abstracts, and explicitly position the use case as (a) early risk prediction vs (b) retrospective risk adjustment. Add sensitivity analyses in Sec. 3.4 that restrict diagnosis inputs to plausibly admission-known subsets (e.g., principal diagnosis only; principal + POA=$'Y'$ only; or excluding diagnoses with POA≠$'Y'$/missing) and report how mortality/PLOS performance changes. Discuss coding/documentation processes and how they could bias NUM_POA_Y_DIAGNOSES (Sec. 4).

2. **Mortality downstream performance discrepancy strongly suggests a pipeline/implementation problem: the Transformer proxy model reports moderate PR-AUC on validation (~ 0.37 in Sec. 3.3), yet downstream "Attention" models show extremely low test PR-AUC (≈ 0.036–0.056 in Sec. 3.4), worse than the non-diagnosis baseline. Such a large drop is atypical if embeddings preserve useful signal and raises concerns about misaligned embeddings/labels, split contamination, masking/padding effects in pooling, feature scaling/concatenation issues, or accidental train/test mix-ups (Sec. 2.4–2.7, Sec. 3.3–3.4).**

   Recommendation: Add explicit sanity checks and diagnostics: (i) train a simple classifier using *only* the frozen patient embedding to predict mortality on the same split and compare its AUC-ROC/PR-AUC to the proxy head (Sec. 3.3–3.4); (ii) verify em-

bedding–label alignment after joins/merges (hash/ID checks) and confirm embeddings are generated separately for train/val/test using the Transformer fit only on the training split (Sec. 2.7); (iii) report embedding norms, fraction of near-constant dimensions, and whether PAD tokens are masked in attention and pooling (Sec. 2.4.3); (iv) ensure downstream preprocessing (scaling/encoding) is fit on training only and applied to val/test. If a bug is found, re-run Sec. 3.4 and update conclusions accordingly.

3. **Data flow and splitting strategy are ambiguous across proxy training and downstream tasks, making it difficult to assess leakage and fairness. Sec. 3.1 mentions a 70/15/15 split on the 1% subsample, but it is unclear whether this split is used for (a) Transformer proxy training, (b) downstream mortality/PLOS training, and (c) baseline feature selection (top-N diagnoses, target encoding). Without a single consolidated description, readers cannot verify that (i) the Transformer never "sees" validation/test labels or examples during representation learning, and (ii) feature engineering (top-N selection, target encoding) is fit only on the training portion (Sec. 2.4–2.7, Sec. 3.1–3.4).**

   Recommendation: Add a single schematic/table (Sec. 2.7) specifying for each stage (Transformer proxy, embedding generation, downstream LR/XGBoost, Baseline 2 code selection, target encoding): exact split sizes, stratification, and what is fit on train vs applied to val/test. Explicitly state that (1) embeddings for val/test are generated by a Transformer trained only on the training subset; (2) top-N code lists are derived from training only (Sec. 2.6.2); and (3) target encoding maps (PAT_COUNTY, ZIP3) are learned on training only, with smoothing, and then applied to val/test.

4. **Severe subsampling (1% / 31,!102 records) is a central experimental constraint, yet conclusions sometimes read as general statements about attention models vs feature engineering rather than about this resource-constrained regime. Given Transformer capacity, training on 31k examples for 3 epochs may be underpowered; conversely, strong tabular baselines may improve substantially when trained on the full 3.1M records. Without any scaling experiment, it is unclear whether the negative result reflects the architecture, the training regime, or sample size (Sec. 2.1, Sec. 3.3–3.4, Sec. 4).**

   Recommendation: Make the limited-scale nature explicit in Sec. 2.1 and Sec. 4 and temper general claims accordingly. If feasible, train Baseline 1 and Baseline 2 on the full dataset to provide a realistic reference point. Add at least a small scaling study for the Transformer (e.g., 1% vs 2% vs 5% subsamples, or a smaller d_{model} / fewer layers enabling more data) with downstream results in Sec. 3.4, or include a quantitative discussion (parameter counts, training steps, expected sample complexity) if additional experiments are infeasible.

5. **Transformer modeling choices are under-specified and, in places, internally ambiguous: ordering of diagnoses (set vs sequence), positional encodings, padding masks in attention and mean pooling, and the dimensionality path from concatenated diagnosis+POA embeddings to the Transformer d_{model} and final 160-D patient embedding are not fully defined. These details are crucial both for reproducibility and for interpreting why a Transformer would help on essentially set-like inputs (Sec. 2.3.2, Sec. 2.4.1–2.4.3).**

   Recommendation: In Sec. 2.4.1–2.4.3, fully specify: (i) the token ordering rule (e.g., principal first, then other diagnoses in file order OTH1..OTH24) and whether this order has semantics; (ii) whether positional embeddings are used and of what type; (iii) attention masking and masked-mean pooling over non-PAD tokens; and (iv) the exact dimensions (D_{dx}, D_{poa}, concatenation dimension, any projection layer, d_{model}, and the mapping to the 160-D pooled embedding). Consider adding a permutation-invariant baseline (e.g., DeepSets / Set Transformer / attention pooling) or justify why sequence modeling is appropriate here.

6. **Proxy-task design is not well justified and may be mismatched to downstream evaluation: the encoder is trained only on mortality (proxy) and then used for both mortality and PLOS, but the paper does not test PLOS-trained or multi-task encoders, nor does it compare frozen embeddings vs end-to-end fine-tuning. For mortality, the two-stage "pretrain on mortality then train another model for mortality" pipeline is not clearly motivated relative to a single end-to-end model (Sec. 2.4, Sec. 3.3–3.4).**

   Recommendation: Add experiments (as feasible) training (i) a PLOS-proxy encoder, (ii) a multi-task encoder (mortality+PLOS heads), and (iii) at least one end-to-end fine-tuning setup where the encoder is updated for the downstream task (Sec. 3.4). If compute prohibits this, expand Sec. 4 to explicitly acknowledge that the negative result may be specific to mortality-proxy + frozen-embedding transfer and may not generalize to end-to-end or task-aligned training.

7. **Baseline 2 is central to the conclusions but is under-specified, and fairness of the comparison is unclear: the value of N is inconsistent/vague ("200–300" vs fixed $N = 200$), selection criteria may leak information if computed on all data, and it is unclear whether principal and POA=$'Y'$ top-N lists are separate, how overlaps are handled, and whether principal diagnoses are included in the POA=$'Y'$ multi-hot (Sec. 2.6.2, Sec. 3.4).**

   Recommendation: Rewrite Sec. 2.6.2 to precisely define Baseline 2: fixed N (or a small tuned set), separate vs joint vocabularies for principal and POA=$'Y'$, whether principal is included in POA=$'Y'$, handling of overlaps/ties, and explicit statement that ranking/frequency counts use training data only. Consider adding a slightly stronger

but still "simple" baseline that uses dx3×POA cross-features (e.g., hashing) to more directly mirror what the Transformer could represent, and report its performance in Sec. 3.4.

8. **Evaluation lacks uncertainty, calibration, and operating-point reporting, limiting the strength and practical meaning of model comparisons—especially when differences are modest for PLOS. The manuscript also lists many metrics (F1, specificity, Brier score) but does not systematically report them (Sec. 2.7, Sec. 3.4).**

   Recommendation: In Sec. 3.4, add bootstrap 95% CIs for AUC-ROC and PR-AUC (and optionally for PR-AUC differences) and indicate which pairwise differences are statistically meaningful. Add calibration evaluation (e.g., reliability plot + Brier score or ECE) at least for the best baseline vs attention model. If claiming admission-time triage utility, report clinically interpretable operating points (e.g., precision at fixed recall) alongside PR-AUC. Align Sec. 2.7 with what is actually reported, or move the full metric table to an appendix.

Minor issues

1. Related work and positioning are thin and not organized into a dedicated section, which makes it hard to assess novelty and how the negative result relates to prior work on diagnosis-code Transformers, set encoders, and administrative-data risk modeling (Sec. 1–2).

   Recommendation: Add a Related Work subsection (end of Sec. 1 or new Sec. 2.x) covering: (i) mortality/LOS prediction from claims/EHR; (ii) code-embedding and Transformer/RNN/set-based models for diagnosis sequences/sets; (iii) uses of POA and missingness in administrative modeling. In Sec. 4, explicitly contextualize why deep models may or may not help in this feature regime.

2. POA category/vocabulary accounting is inconsistent: the manuscript describes "six categories" but elsewhere includes PAD_POA while still calling it six; the raw-to-processed mapping (Y/N/U/W/1/blank/invalid) is not enumerated in one place (Sec. 2.3.2).

   Recommendation: Provide a single table in Sec. 2.3.2 mapping raw POA values (including blank/null/invalid) to processed categories, and clearly distinguish clinical categories vs special tokens (PAD). State the exact POA vocabulary size used by the embedding layer.

3. Non-diagnostic feature preprocessing and target encoding are under-specified, and leakage controls are not explicit. This includes age bin boundaries, final category groupings for RACE/ETHNICITY/SEX/TYPE_OF_ADMISSION, and the precise target-encoding formula and whether encodings are outcome-specific (mortality vs PLOS) (Sec. 2.3.1).

Recommendation: Add a concise table (main text or appendix) with final categories and bin boundaries, and specify the target-encoding estimator (smoothing, prior) and whether it is learned separately per outcome. Explicitly state that target encoders are fit on training only and applied to validation/test.

4. Ablations are missing, making it difficult to tell whether underperformance is due to attention per se vs other design choices (3-character truncation, explicit POA embedding, POA_M_MISSING handling, mean pooling).

   Recommendation: Add a minimal ablation set (at least on PLOS): (i) diagnoses-only (no POA), (ii) POA with missing merged into "unknown," (iii) alternative pooling ([CLS] token or learned attention pooling). Report in Sec. 3.4 and discuss in Sec. 4.

5. Quantification of POA distributions (especially missing/invalid POA) is not systematic, despite POA_M_MISSING being central to the method and interpretation (Sec. 2.3.2, Sec. 3.1–3.2).

   Recommendation: Add a table in Sec. 3.1/3.2 reporting proportions of each processed POA category overall and stratified by principal vs secondary diagnoses, and briefly discuss implications for model learning and bias.

6. Figures are numerous and often difficult to interpret due to inconsistent labeling/normalization, unclear abbreviations, small fonts/low resolution, and missing sample-size annotations (e.g., Figures 1–7, 9–11, 16–21). Some plots appear potentially erroneous (e.g., a figure where top categories all have frequency $\mathbf{1.0}$).

   Recommendation: Standardize figure formatting: explicit units (count vs proportion), include $\boldsymbol{n}$ in captions, decode abbreviations, order categories logically, and export at $\geq \mathbf{300}$ dpi or vector. Audit plots for normalization errors (especially those with suspiciously uniform frequencies) and correct as needed.

7. Interpretability section notes attention visualization was "unsuccessful" without enough methodological detail, and SHAP discussion could more clearly connect top drivers to POA/diagnosis patterns (Sec. 3.5).

   Recommendation: Describe exactly what attention visualizations were attempted (which layers/heads, per-patient vs aggregated) and what failure mode occurred (diffuse/unstable attention, etc.). Use **2–3** concrete examples from SHAP (specific diagnoses/POA patterns) with brief clinical interpretation, and optionally test token ablation (remove a code and measure probability change) as an alternative to attention-as-explanation.

8. Limitations and ethical/fairness considerations are present but scattered, and do not clearly enumerate representativeness (Texas-only, single year), lack of external validation, and potential disparate impact across demographic groups (Sec. 4).

Recommendation: Add a dedicated Limitations subsection in Sec. 4 that systematically lists: discharge-abstract temporal limitations, coding/documentation bias, subsampling/compute, lack of external validation, and representativeness. Add a short paragraph on fairness/privacy (even if only as future work), especially given use of geography encodings (ZIP3/county).

Very minor issues

1. Small internal inconsistencies in counts/rounding (e.g., **31,!102** described as "1%" of **3,!110,!296**; split counts not matching **70/15/15** rounding) and minor naming inconsistencies for NUM_POA_Y_DIAGNOSES_scaled across text/figures (Sec. 2.1, Sec. 3.1, Sec. 3.5).

   Recommendation: Report exact achieved percentages for subsampling and splits, or describe the deterministic rounding rule. Standardize the feature name (scaled vs unscaled) across Methods, Results, and figure labels/captions.

2. Typographical/formatting issues: inconsistent capitalization/quoting for POA categories, mixed heading styles/Markdown artifacts, occasional line-break artifacts splitting words, and inconsistent use of "non-diagnosis/nondiagnosis" (various locations).

   Recommendation: Proofread and standardize terminology, headings, and quoting style; remove Markdown artifacts and fix line-break/word-splitting issues prior to final submission.

3. Edge-case specification gaps in sequence construction (e.g., what happens if principal diagnosis is null/empty) and UNK-handling details for rare ICD-3 codes are not stated (Sec. 2.3.2).

   Recommendation: Add one or two sentences specifying handling of null principal diagnosis (if any) and the thresholding/mass for UNK (how many codes mapped to UNK; effect on vocabulary size).

Key statements and references

- • **The study used the 2018 Texas Hospital Inpatient Discharge Public Use Data File (PUDF), an administrative dataset containing 3,!110,!296 inpatient discharge records from Texas hospitals with demographics, admission/discharge details, ICD-10-CM diagnosis and procedure codes, and Present On Admission (POA) indicators, and included all records after removal of a single entirely null column.**
- *Reference(s):* 2018 Texas Hospital Inpatient Discharge Public Use Data File
- • **In-hospital mortality was operationalized as a binary outcome where MORTALITY $= 1$ if the Patient Discharge Disposition (PAT_STATUS) equaled the code "20" (Expired), corresponding to death during hospital-**

ization, and 0 otherwise, yielding a mortality rate of 1.74% in this statewide administrative dataset.

- *Reference(s):* PAT_STATUS

- **• Prolonged Length of Stay (PLOS) was defined as a binary variable using the 75th percentile of \text{LENGTH_OF_STAY} (6 days) as the threshold, such that PLOS = 1 if \text{LENGTH_OF_STAY} > 6 days and 0 otherwise, with records missing \text{LENGTH_OF_STAY} (0.003%) excluded from this task, resulting in approximately 21.1% of admissions classified as PLOS.**

- *Reference(s):* LENGTH_OF_STAY

- **• For in-hospital mortality prediction on the held-out test set, XGBoost models using the Baseline 2 feature set (non-diagnostic features plus simple diagnosis encodings of the top 200 principal diagnoses and top 200 POA=$'Y'$ diagnoses) achieved the highest Area Under the Precision-Recall Curve (AUC-PR) of 0.266, whereas XGBoost models using the attention-based diagnosis embeddings achieved an AUC-PR of only 0.036, indicating that the simpler explicit diagnosis encoding substantially outperformed the learned attention-based representations in this setting.**

- *Reference(s):* XGBoost, Baseline 2, POA=$'Y'$

Mathematical consistency audit

This section audits **symbolic/analytic** mathematical consistency (algebra, derivations, dimensional/unit checks, definition consistency).

Maths relevance: light

The paper contains minimal formal mathematics (no numbered equations/derivations). The key analytic content is the definition and consistency of categorical vocabularies (diagnosis and POA), embedding dimensionalities (D_{dx}, D_{poa}, pooled patient embedding size), sequence/padding handling, and feature construction (counts, standardization, target encoding). The main internal-consistency problems are definitional: mismatched POA vocabulary cardinality and missing detail needed to verify dimensional consistency and pooling under padding.

Checked items

1. ✔ **POA processed categories = 6 (clinical categories + explicit missing)** (Sec. 2.3.2, p.3)
    - **Claim:** Raw POA statuses are standardized into six categories: Y, N, U, W, 1 (Exempt), and POA_M_MISSING (for blank/null POA when a diagnosis is present).
    - **Checks:** definition consistency, category cardinality

- **Verdict:** PASS; confidence: high; impact: minor
- **Assumptions/inputs:** A diagnosis code may be present while its POA field is blank/null., Invalid POA codes are mapped to POA_M_MISSING.
- **Notes:** This section gives a self-contained and internally consistent definition of six processed POA categories, including a distinct missingness category conditional on diagnosis presence.

2. ✖ **POA vocabulary cardinality mismatch (6 vs 7 with PAD)** (Sec. 3.2, p.6–7 (Diagnosis_POA vocabulary paragraph))
    - **Claim:** The POA status vocabulary comprised **6** categories, then lists: POA_Y, POA_W_CLIN, POA_U_DOC, POA_N, POA_1_EXEMPT, POA_M_MISSING, and PAD_POA.
    - **Checks:** definition consistency, counting/cardinality, notation consistency across sections
    - **Verdict:** FAIL; confidence: high; impact: moderate
    - **Assumptions/inputs:** PAD_POA is a special token used for padding sequences., The listed items are intended to be the full POA embedding vocabulary.
    - **Notes:** The text states "**6** categories" but enumerates **7** distinct tokens when PAD_POA is included. This conflicts with Sec. 2.3.2/2.4.1, which define/create a six-category POA vocabulary (without clarifying whether special tokens are included).

3. ✔ **Diagnosis vocabulary includes special tokens** (Sec. 2.4.1, p.3–4 and Sec. 3.2, p.6)
    - **Claim:** Rare diagnosis codes map to UNK_DX; sequences are padded (implying a PAD_DX token) and Results report a diagnosis vocab including UNK_DX and PAD_DX.
    - **Checks:** definition consistency, special-token consistency
    - **Verdict:** PASS; confidence: medium; impact: minor
    - **Assumptions/inputs:** Padding uses a dedicated diagnosis token PAD_DX., UNK_DX is used only for low-frequency/unknown codes.
    - **Notes:** The inclusion of UNK_DX and PAD_DX is consistent with frequency-thresholding and fixed-length padding. Methods do not explicitly name PAD_DX, but padding to length **25** implies some pad symbol is required.

4. ✔ **Token embedding dimension by concatenation** (Sec. 2.4.1, p.3–4)
    - **Claim:** A diagnosis embedding in $\mathbb{R}^{D_{\text{dx}}}$ concatenated with a POA embedding in $\mathbb{R}^{D_{\text{poa}}}$ yields a token vector in $\mathbb{R}^{D_{\text{dx}}+D_{\text{poa}}}$.
    - **Checks:** dimensional consistency
    - **Verdict:** PASS; confidence: high; impact: minor

- **Assumptions/inputs:** Concatenation is along the feature/channel dimension.
- **Notes:** Concatenation implies additive dimensionality; the stated result $D_{\text{dx}} + D_{\text{poa}}$ is correct.

5. ⚠ **Transformer model dimension not specified relative to concatenated embeddings** (Sec. 2.4.1–2.4.3, p.3–4; Sec. 3.3, p.7)
   - **Claim:** Concatenated embeddings are fed into a Transformer encoder, and the resulting pooled patient embedding is reported as **160**-dimensional.
   - **Checks:** dimensional consistency, missing definitional steps
   - **Verdict:** UNCERTAIN; confidence: medium; impact: critical
   - **Assumptions/inputs:** Transformer requires a fixed internal representation size (model dimension)., The pooled vector has the same dimension as the Transformer output per token.
   - **Notes:** The paper does not specify whether the Transformer's d_{model} equals $D_{\text{dx}} + D_{\text{poa}}$ or whether a projection is applied. Without this, the mapping from concatenated token vectors to a **160**-D patient embedding cannot be verified.

6. ✔ **Sequence length and padding constraint** (Sec. 2.4.2, p.4; Sec. 3.2, p.6)
   - **Claim:** Sequences are padded to a maximum length of **25** tokens, corresponding to principal diagnosis plus **24** other diagnoses.
   - **Checks:** constraint consistency, definition consistency
   - **Verdict:** PASS; confidence: high; impact: minor
   - **Assumptions/inputs:** There are at most **24** other diagnosis fields in the dataset.
   - **Notes:** The maximum length **25** is consistently justified and reused.

7. ⚠ **Pooling operation definition (mean over sequence) and padding mask** (Sec. 2.4.3, p.4)
   - **Claim:** A single patient representation is obtained by mean pooling Transformer outputs across the sequence dimension.
   - **Checks:** definition completeness, invariance/sanity under padding
   - **Verdict:** UNCERTAIN; confidence: high; impact: critical
   - **Assumptions/inputs:** Sequences are padded to length **25** for batching.
   - **Notes:** Mean pooling is not specified as masked or unmasked. If unmasked, representations depend on the number of PAD tokens (and on PAD embedding values), changing the mathematical definition and comparability across patients with different numbers of diagnoses.

8. ⚠ **Diagnosis–POA pair inclusion rule completeness** (Sec. 2.3.2, p.3)

- **Claim:** The set includes (principal diagnosis, principal POA) and includes each other diagnosis j only if its diagnosis code is not null; pairs with null diagnosis codes are excluded.
- **Checks:** definition completeness, edge-case consistency
- **Verdict:** UNCERTAIN; confidence: medium; impact: minor
- **Assumptions/inputs:** Principal diagnosis is typically present, but could in principle be null/empty.
- **Notes:** Other-diagnosis inclusion is explicitly conditional on non-null codes; the principal diagnosis inclusion is not stated as conditional. If principal codes can be null/empty, the construction rule is incomplete.

9. ✔ **Engineered count feature NUM_POA_Y_DIAGNOSES** (Sec. 2.3.1, p.3)
   - **Claim:** NUM_POA_Y_DIAGNOSES counts the total number of (principal and other) diagnoses explicitly marked POA=$'Y'$ for each patient and is standardized with z-score normalization.
   - **Checks:** definition consistency, constraint sanity
   - **Verdict:** PASS; confidence: high; impact: minor
   - **Assumptions/inputs:** Each diagnosis field has an associated processed POA category.
   - **Notes:** The count definition is clear and bounded by the number of diagnosis slots. Later references to NUM_POA_Y_DIAGNOSES_scaled are consistent with z-score standardization.

10. ⚠ **Target encoding definition depends on outcome (two-task ambiguity)** (Sec. 2.3.1, p.3)
    - **Claim:** PAT_COUNTY and ZIP3 are target-encoded using "outcome prevalence within each category" with smoothing on training data.
    - **Checks:** definition completeness, symbol/target consistency
    - **Verdict:** UNCERTAIN; confidence: high; impact: moderate
    - **Assumptions/inputs:** There are two outcomes (Mortality, PLOS) modeled separately.
    - **Notes:** Because there are two separate prediction targets, the phrase "outcome prevalence" is ambiguous: either separate encodings per outcome are required or a single encoding is reused. The mathematical definition of these features is therefore not fully specified.

11. ✔ **Confusion matrix totals consistent with stated test size** (Sec. 3.4 and Fig. 15 caption, p.8–9; split sizes in Sec. 3.1, p.5)
    - **Claim:** The PLOS confusion matrix counts sum to the test set size $(\mathbf{4,!666})$ reported for the $\mathbf{1\%}$ subsample split.
    - **Checks:** internal arithmetic consistency, cross-reference consistency
    - **Verdict:** PASS; confidence: high; impact: minor

- **Assumptions/inputs:** The confusion matrix corresponds to the held-out test set of size $4,!666$ for the subsample.
- **Notes:** Counts $3,!478 + 187 + 588 + 413 = 4,!666$, matching the stated test size for the subsample split, so these two parts are internally consistent.

Limitations

- The provided paper text contains essentially no formal equations/derivations; the audit is therefore limited to checking definitional and dimensional consistency of the described modeling pipeline.
- Many mathematically relevant implementation choices (masking in pooling, exact positional encoding usage, exact d_{model}, and target-encoding formula) are not specified; where verification depends on these, items are marked UNCERTAIN rather than inferred.
- Figures are referenced for distributions/curves but do not contribute additional symbolic definitions needed to verify the core mathematical consistency beyond what is in the text.

Numerical results audit

This section audits **numerical/empirical** consistency: reported metrics, experimental design, baseline comparisons, statistical evidence, leakage risks, and reproducibility.

Eight numerical checks were run: 4 PASS and 4 FAIL. Failures were driven by (i) an internal count inconsistency in the POA vocabulary listing, (ii) small rounding/proportion mismatches in the stated 1% subsample size and $70/15/15$ split counts, and (iii) a small difference between a dataset-wide PLOS prevalence claim and the prevalence implied by the test confusion matrix (though still close in absolute terms).

Checked items

1. ✖ **C1_dataset_subsample_size** (Page 5 (Section 3.1))
    - **Claim:** "A 1% random subsample of $31,!102$ records was extracted…" from a raw dataset of $3,!110,!296$ records.
    - **Checks:** percentage_of_total
    - **Verdict:** FAIL
    - **Notes:** Expected $\text{round}(3,!110,!296 \times 1\%) = 31,!103$ vs stated $31,!102$ (off by 1). Implied percent $= 31,!102/3,!110,!296 = 0.0099996913$ (absolute diff vs 0.01 is 3.0865×10^{-7}). Verdict reflects strict rel_tol setting despite small absolute discrepancy.
2. ✔ **C2_split_counts_sum_to_subsample** (Page 5 (Section 3.1))
    - **Claim:** Subsample of $31,!102$ split into 70% training ($21,!770$), 15% validation ($4,!666$), and 15% test ($4,!666$).
    - **Checks:** parts_sum_to_total

- **Verdict:** PASS
- **Notes:** Counts sum exactly: $21,!770 + 4,!666 + 4,!666 = 31,!102$.

3. ✔ **C3_split_percentages_sum_to_100** (Page 5 (Section 3.1))
   - **Claim:** Split described as $70\%/15\%/15\%$.
   - **Checks:** percentages_sum_to_100
   - **Verdict:** PASS
   - **Notes:** Percentages sum exactly: $70 + 15 + 15 = 100$.
4. ✖ **C4_split_counts_match_claimed_percentages** (Page 5 (Section 3.1))
   - **Claim:** Split counts ($21,!770$ / $4,!666$ / $4,!666$) are described as $70\%/15\%/15\%$ of $31,!102$.
   - **Checks:** counts_vs_percentages
   - **Verdict:** FAIL
   - **Notes:** Raw expectations: $31,!102 \times 0.7 = 21,!771.4$; $\times 0.15 = 4,!665.3$. Rounded expected counts: train $21,!771$; val $4,!665$; test $4,!665$ vs stated $21,!770/4,!666/4,!666$ (each differs by 1). Implied proportions from stated counts are close (train 0.699955; val/test 0.1500225), but the check flagged due to the count differences.
5. ✖ **C5_poa_vocab_count_internal_consistency** (Page 6 (Section 3.2))
   - **Claim:** "The POA status vocabulary comprised 6 categories (…) and 'PAD_POA'."
   - **Checks:** enumeration_count_consistency
   - **Verdict:** FAIL
   - **Notes:** Claimed size 6 conflicts with 7 listed items. This could be an off-by-one error or an implicit convention that PAD_POA is excluded from the count.
6. ✔ **C6_max_sequence_length_matches_diagnosis_slots** (Page 4 (Section 2.4.2) and Page 6 (Section 3.2))
   - **Claim:** Maximum length is 25 tokens, corresponding to principal diagnosis plus 24 other diagnoses.
   - **Checks:** simple_arithmetic_identity
   - **Verdict:** PASS
   - **Notes:** Identity holds exactly: $1 + 24 = 25$.
7. ✔ **C7_confusion_matrix_total_equals_test_size** (Page 9 (Figure 15 caption) and Page 5 (Section 3.1))
   - **Claim:** Confusion matrix counts for best PLOS model: TN= $3,!478$, FP= 187, FN= 588, TP= 413; test set size is $4,!666$.
   - **Checks:** parts_sum_to_total

- **Verdict:** PASS
- **Notes:** Counts sum exactly to the stated test size: $3,!478 + 187 + 588 + 413 = 4,!666$.

8. ✖ **C8_pLOS_prevalence_from_confusion_matrix** (Page 9 (Figure 15 caption) and Page 5 (Section 3.1))
    - **Claim:** PLOS affects approximately 21.1% of admissions; in the test confusion matrix, actual positives are FN+TP and total is $4,!666$.
    - **Checks:** rate_from_counts
    - **Verdict:** FAIL
    - **Notes:** From the test confusion matrix, prevalence $= (588 + 413)/4666 = 0.2145306$ (21.453%), differing from 0.211 by 0.00353 (0.353 percentage points). Despite being within the stated abs_tol, the check was marked FAIL in exec output; interpret as a small discrepancy and potential scope mismatch (dataset-wide vs test-set).

Limitations

- Only parsed text and figure captions were available; no raw datasets, intermediate tables, or model outputs were provided to recompute most performance metrics or descriptive statistics.
- No checks rely on extracting numeric values from plotted axes/pixels; therefore distribution-based claims shown only in figures (without explicit numbers) cannot be verified.
- Some numerical statements are definitional (e.g., mapping rules) rather than computational and thus offer limited scope for FAST numerical verification beyond simple arithmetic and count consistency.