Sparse Category Routing and Fairness-Aware Optimization for Medical Decision Extraction

Ahmed Elshehaby*, **Mohamed Abdalla***, **Youssef Mohamed**†
Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI)
{ahmed.elshehaby, mohamed.abdalla, youssef.mohamed}@mbzuai.ac.ae

Abstract

Extracting structured medical decisions from ICU discharge summaries is hard because of long documents, severe category imbalance across nine DICTUM decision types, and a fairness-aware evaluation that penalizes inconsistent performance across demographic subgroups. We present our system for the MedExACT 2026 shared task (Elgaar et al., 2026), which fine-tunes BiomedBERT with a composite loss combining label-smoothed cross-entropy, a soft token-F1 auxiliary term, and R-Drop regularization. At inference time we apply a deterministic ensemble: half-offset sliding-window augmentation across four window configurations, dual-branch logit aggregation from the same checkpoint, per-category length calibration on the Anchor Branch, and sparse routing to a context-weighted specialist branch for Cat7, which has strong span-length evidence, and Cat4, which is included only as a rare-category heuristic. In a single-seed validation run, adding R-Drop improved Overall_F1 by 1.24 points over the CE + soft-F1 baseline, with a larger 1.70-point gain on Worst-Group F1; these gains should be interpreted as single-run observations rather than variance-estimated effects. Our best submission achieves Span F1 of 0.4900, Token F1 of 0.6796, and an official Overall_F1 of 0.5724, with the African American subgroup as the Worst-Group bottleneck at Base_Score 0.5601.

1 Introduction

ICU clinicians record dozens of decisions per patient in discharge summaries, from drug orders and referrals to test-result evaluations and disposition plans. The DICTUM taxonomy (Ofstad et al., 2016) defines ten decision types. The MedDec dataset (Elgaar et al., 2024) provides span-level annotations over 451 de-identified MIMIC-III discharge summaries (Johnson et al., 2016) with inter-annotator agreement of Cohen's $\kappa = 0.74$, and the MedExACT 2026 shared task restricts scoring to nine of these as in-scope (Categories 1–9). The MedExACT 2026 shared task (Elgaar et al., 2026) challenges participants to extract these spans and classify each into one of the nine categories under a fairness-aware metric that penalizes inconsistent performance across demographic subgroups.

The task is harder than standard clinical NER for three reasons. Notes average 1,571 whitespace tokens, well beyond the 512-token context of standard BERT encoders. The nine in-scope shared-task categories are severely imbalanced in the official 350-document training split: Category 8 (Deferment) has only 84 annotated spans, whereas Category 3 (Defining problem) has 17,015. And the ranking metric (Overall_F1) is the average of aggregate Base_Score and Worst-Group F1, so improving the aggregate at the expense of any single demographic subgroup directly costs points.

We describe a system based on a single fine-tuned BiomedBERT checkpoint (Figure 1). Training uses a composite loss that adds soft token-F1 and R-Drop regularization to label-smoothed cross-entropy. Inference applies four half-offset sliding-window views to reduce boundary truncation, filters Anchor-branch predictions for three categories by refined span length, and then replaces Cat7 and Cat4 predictions with a context-weighted specialist branch. Cat7 routing is supported by span-length analysis, while Cat4 routing is a rare-category heuristic with weak distributional evidence. The system achieves Overall_F1 of 0.5724 on the official test set.

*Equal contribution.
†Corresponding author.

Proceedings of the BioNLP 2026 (Shared Tasks), pages 201–212

2 Related Work

Clinical NER and span extraction. Prior work on MedDec established that fine-tuned token classifiers far outperform zero-shot LLMs: MedDecXtract (Elgaar et al., 2025) reached 34.8 span F1 with fine-tuned RoBERTa vs. 4.8 for one-shot Llama-3.1-8B. BIO labeling over pretrained transformers is the standard approach for flat clinical NER (Gu et al., 2022), with domain-specific pretraining consistently outperforming general encoders. SpanNER (Fu et al., 2021) and the boundary-smoothing loss of Zhu and Li (2022) point to span-level and boundary-aware modeling as a more direct fit for this task: among detected spans, the dominant failure mode is boundary drift rather than category confusion (Section 7). Overlap-capable extraction (Li et al., 2021) would also apply since many labeled MedDec documents contain overlapping spans, but BIO tagging cannot represent these.

Long-document inference. ERNIE-Doc (Ding et al., 2021) addresses full-document context via retrospective attention. Practical sliding-window inference with half-offset TTA was the most reliable option in our setting.

Regularization and fairness-aware selection. R-Drop (Liang et al., 2021) was our single largest training gain in the validation run. Group DRO (Sagawa et al., 2020) and Just Train Twice (Liu et al., 2021) target training-time worst-group robustness; Group DRO destabilized training in our setting (proxy span F1: 15.4 vs. 39.9 baseline), so our fairness-aware contribution is limited to selecting checkpoints and post-processing settings by Overall_F1 rather than introducing an effective training-time fairness method.

3 Task and Evaluation

Given a full ICU discharge summary, systems must detect contiguous text spans expressing medical decisions and assign each span one of nine in-scope DICTUM categories (Elgaar et al., 2026). The evaluation combines performance and subgroup robustness. The Base_Score averages span- and token-level F1:

$$\text{Base\_Score} = \frac{\text{Span F1} + \text{Token F1}}{2} \quad (1)$$

Worst-Group F1 is the minimum subgroup Base_Score across nine demographic partitions (sex: Female/Male; race: White, African American, Hispanic, Asian, Other; language: English/Non-English):

$$\text{Worst-Group F1} = \min\left(\{\text{Base\_Score}_i\}_{i=0}^{8}\right) \quad (2)$$

The final ranking metric is:

$$\text{Overall\_F1} = \frac{\text{Base\_Score} + \text{Worst-Group F1}}{2} \quad (3)$$

A model that improves aggregate Base_Score while degrading any subgroup scores lower on Overall_F1 than one that simply holds the worst group steady.

Encoder	P	R	F1
BiomedBERT fulltext	36.2	42.8	**39.2**
BioClinical-ModernBERT-lg	34.7	44.7	39.1
BiomedBERT abstract	36.0	42.6	39.0
BioClinical-ModernBERT-base	33.3	42.2	37.2
SciBERT	33.8	41.2	37.1
Bio Discharge Summary BERT	33.8	40.7	36.9
BlueBERT	31.5	40.4	35.4
Bio-GottBERT	30.2	39.9	34.4
Clinical_ModernBERT	10.6	22.4	14.4

Table 1: Backbone comparison (span-level precision, recall, F1 on proxy scorer, 5,000 training steps, seed 42).

4 System Description

4.1 Backbone and Task Formulation

We treat medical decision extraction as BIO token labeling, assigning each token a label from $\{B\text{-}k, I\text{-}k, O\}$ where $k \in \{1, \ldots, 9\}$, yielding 19 labels. We use BiomedNLP-BiomedBERT-base-uncased-abstract-fulltext (Gu et al., 2022) as the encoder, pre-trained on both PubMed abstracts and full texts. In a preliminary comparison across multiple biomedical and clinical encoders (Table 1), BiomedBERT fulltext produced the highest span F1 (39.2%) and was selected as the foundation for all subsequent experiments.

A BIO-constrained Viterbi decoder enforces legal tag transitions at inference time: the start state allows only O or any B-$*$; I-k is permitted only after B-k or I-k (same-category continuation only). BIO tagging is imperfect for this dataset: when two overlapping spans share a token, only one annotation can be represented, so some gold spans are silently dropped during training.

4.2 Text Normalization and Preprocessing

MIMIC-III discharge summaries contain formatting artifacts that affect both tokenization and span

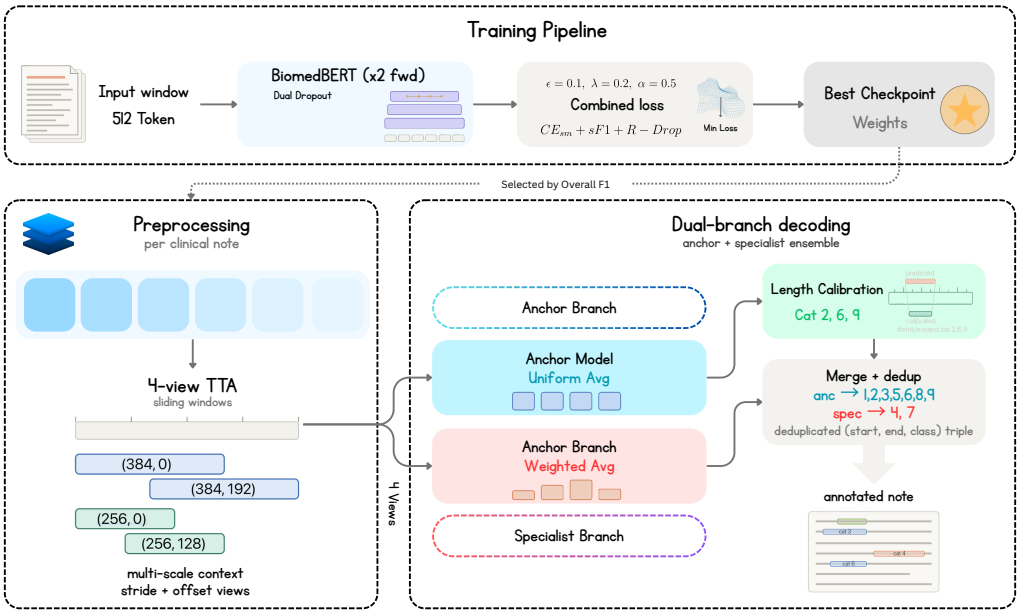


Figure 1: System pipeline. **Training:** 512-token windows (biased toward annotated regions) pass twice through BiomedBERT under independent dropout. A composite loss is backpropagated for 5500 steps; the best checkpoint by Overall_F1 is kept. **Inference:** four half-offset sliding-window views feed an Anchor Branch (uniform avg) and a Specialist Branch (weighted avg). Length calibration is applied to the Anchor Branch only, filtering predictions for Categories 2, 6, and 9 by refined character length. The merge step keeps Categories 1, 2, 3, 5, 6, 8, 9 from the calibrated Anchor Branch and substitutes Categories 4 and 7 from the Specialist Branch, followed by deduplication over (s, e, cat) triples.

boundary alignment. We apply a three-step preprocessing pipeline before training.

Character normalization. Unicode whitespace variants (non-breaking spaces `U+00A0`, figure spaces `U+2007`, narrow no-break spaces `U+202F`, BOM `U+FEFF`) are mapped to ASCII space. Typographic quotes (`U+2018`/`U+2019`/`U+201C`/`U+201D`) are folded to ASCII equivalents, and en-dashes, em-dashes, and minus signs (`U+2013`/`U+2014`/`U+2212`) are replaced with hyphens. Bullets (`U+2022`) are likewise mapped to hyphens. All replacements are single-character to single-character so that character offsets are preserved without remapping.

Boilerplate filtering. An extractive line-level filter removes administrative headers (*Admission Date*, *Discharge Date*, *Sex*, *Service*, *Attending*, etc.) and section markers (*Discharge Medications*, *Discharge Disposition*, *Followup Instructions*) that never contain decision spans. Blank lines and decorative separators (___, −, ===) are also removed. Lines that overlap any gold annotation span are always kept, regardless of boilerplate matching, so no annotated text is lost. A bidirectional character-level offset map tracks the mapping between original and shortened documents, allowing predictions on the shortened text to be projected back to original offsets for evaluation.

Whitespace compaction. Runs of consecutive spaces or newlines that do not overlap an annotation span are collapsed: multi-space runs become a single space, and runs containing a newline become a single newline. Runs within annotation spans are left untouched to preserve byte-identical span text. This step reduces document length by 8–15% on average without affecting any gold span content.

Span refinement at evaluation. The official evaluator applies a `refine_span` routine during matching, which expands boundaries that cut through an alphanumeric word and strips leading and trailing punctuation tokens using NLTK's `TreebankWordTokenizer`, preserving de-identified PHI markers (`[**...**]`). In the final routed inference recipe, this routine was used only to compute refined character length for category-specific filtering (Cat2, Cat6, Cat9) on the Anchor Branch; the final submitted offsets remained the

original decoded span boundaries.

4.3 Training Objective

The composite loss function is:

$$\mathcal{L} = \mathcal{L}_{\text{CE}}^{\text{smooth}} + \lambda\,\mathcal{L}_{\text{soft-F1}} + \alpha\,\mathcal{L}_{\text{R-Drop}} \qquad (4)$$

with label smoothing ϵ=0.10, λ=0.20, α=0.50.

Label-smoothed cross-entropy. Applied over all BIO positions to reduce overconfidence on the dominant O label, which accounts for >85% of tokens.

Soft token-F1. A differentiable macro soft-F1 over token-category probabilities, computed by summing $P(B\text{-}k) + P(I\text{-}k)$ for each category and forming soft true positives, false positives, and false negatives:

$$\mathcal{L}_{\text{soft-F1}} = 1 - \frac{1}{9}\sum_{k=1}^{9} \frac{2\,\text{TP}_k + \epsilon}{2\,\text{TP}_k + \text{FP}_k + \text{FN}_k + \epsilon} \qquad (5)$$

This matches the training signal to the token-level component of the evaluation metric.

R-Drop. Each window passes through the model twice under independent dropout masks. A symmetric KL divergence is added between the two token-logit distributions on the 19 BIO label positions of valid (non-padding) tokens:

$$\mathcal{L}_{\text{R-Drop}} = \frac{1}{2}\left[\text{KL}(p_1 \| p_2) + \text{KL}(p_2 \| p_1)\right] \qquad (6)$$

where p_1, p_2 are the softmax distributions from the two forward passes. The primary losses are averaged across both passes before backpropagation.

Figure 2 illustrates the interaction of the three loss terms. Training uses online patch sampling with a positive-window bias probability of 0.70, drawing 512-token windows anchored near annotated spans (with ±10-character fallback for tokenizer boundary alignment) rather than feeding full documents. Mixed precision (fp16 autocast/GradScaler) is enabled on CUDA, with gradient clipping at max norm 1.0.

4.4 Long-Document Inference and TTA

A single 512-token sliding window with a fixed stride misses spans that fall near window boundaries, a non-trivial risk given that documents average 1,571 whitespace tokens. We use a four-configuration TTA ensemble with window configurations applied in order: $(384, 0)$, $(384, 192)$, $(256, 0)$, $(256, 128)$, where the notation (w, o) denotes window size w and starting half-offset o. Figure 3 shows how the four views tile across a document. Token logits from overlapping windows within each view are averaged in *logit space* (before softmax and before decoding) at the document level. The four views cover each document position under at least two different boundary alignments, reducing edge-truncation artifacts without retraining.

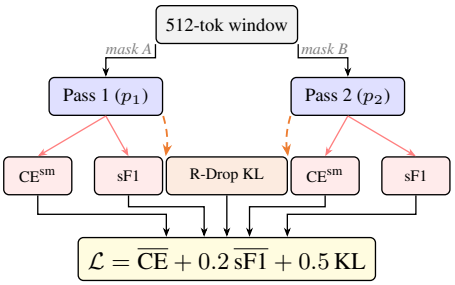


Figure 2: R-Drop training. Each window passes through BiomedBERT twice under different dropout masks. Both passes produce CE and soft-F1 losses; the R-Drop term adds a symmetric KL penalty between p_1 and p_2. Losses are averaged across passes (overbar) before backpropagation.

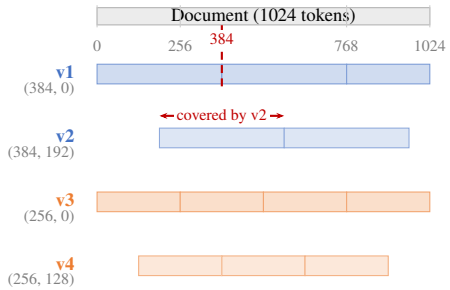


Figure 3: Half-offset TTA tiling for a 1024-token document. Each row shows sliding windows for one view (w, o). Blue: 384-token windows (v1–v2); orange: 256-token windows (v3–v4). The dashed line marks where v1 splits at token 384; v2's half-offset places this position mid-window (red arrow), so every token is covered under at least two different boundary alignments.

4.5 Sparse Category-Routing Ensemble

Statistical analysis of gold character-level span lengths provides a clear routing motivation for Cat7, but not for Cat4. Figure 7 visualizes these distributions. Cat7 (Evaluating test result) has a mean span of 79.9 characters and a median of 60, compared to a median of 28 across the anchor-pool categories 3, 5, and 6. A Mann-Whitney test confirms this separation for Cat7 ($p < 10^{-16}$, rank-

biserial $= -0.40$). Cat4 has much weaker length evidence ($p = 0.21$); we include it in the Specialist Branch only because it is extremely rare (107 training spans, 0.25% of all training annotations) and because development runs suggested that the context-weighted branch could recover occasional Cat4 spans. We therefore treat Cat4 routing as an exploratory low-frequency-category heuristic, not as a statistically justified span-length intervention.

The **Anchor Branch** handles categories 1, 2, 3, 5, 6, 8, and 9 via uniform aggregation over the four TTA views, i.e. view weights $[0.25, 0.25, 0.25, 0.25]$ in logit space. The **Specialist Branch** handles categories 4 and 7 via weighted aggregation with weights $[0.4, 0.4, 0.1, 0.1]$ in view order, biasing toward broader context windows. Broader routing searches over all nine categories consistently overfit the 53-document validation set; two-category routing was the most stable intervention found.

4.6 Character-Level Span-Length Calibration

Before sparse category routing, the Anchor Branch is filtered with category-specific character-length thresholds derived from training span statistics, after which Categories 4 and 7 are replaced by Specialist-branch predictions and the final span set is deduplicated. Thresholds are applied to *refined character length* after a text-boundary repair step that expands boundaries cut through alphanumeric words and strips leading/trailing non-content tokens (using TreebankWordTokenizer), preserving PHI-style `[**...**]` markers. Three categories have constraints, applied only to the Anchor Branch. Thresholds were chosen by a small grid search on the validation split, seeded near the corresponding tail percentile of training-span lengths (p95 for upper bounds, p05 for lower bounds): category 2 drops spans longer than 98 characters (training p95 is 91 chars; the validation optimum sat slightly above this percentile); category 6 drops spans shorter than 3 characters; category 9 drops spans shorter than 24 characters (training p05 is 32 chars; 24 was preferred on validation because it avoided deleting short but valid spans). The final span set is deduplicated over exact (s, e, cat) triples. No overlap suppression or containment deletion is applied, since many labeled documents contain overlapping gold span pairs and blunt deletion damages recall.

4.7 Fairness-Aware Model Selection

We use Overall_F1 as the selection criterion for checkpoint selection, TTA weight tuning, and calibration threshold search. This aligns development decisions with the shared-task metric, so aggregate gains are not preferred when they reduce Worst-Group F1. It does not impose fairness constraints during optimization. Group DRO (Sagawa et al., 2020) was explored during development but achieved only proxy span F1 of 15.4%, most likely because category-level grouping was too coarse. Our fairness-aware component is therefore model and hyperparameter selection by Overall_F1, not a successful training-time fairness method. R-Drop (added for regularization) raises Worst-Group F1 more than Base_Score in our single validation run, by 1.70 vs. 0.77 points (Section 7), but this should be read as an observed robustness gain rather than proof of a fairness-optimized training recipe.

5 Experimental Setup

5.1 Data

The MedDec resource (Elgaar et al., 2024) contains 451 de-identified MIMIC-III discharge summaries (Johnson et al., 2016). For MedExACT, we restrict training and evaluation to the nine in-scope categories (Categories 1–9), excluding Categories 10 and 11 present in the broader MedDec annotations. We use 350 documents for training, 53 for validation, and 48 held-out test documents released by the shared-task organizers for final evaluation. The public release contains 43,640 training and 7,044 validation annotations. Restricting to the nine in-scope shared-task categories yields the 43,418 training spans in Table 2 and 7,022 validation spans; the 222 training annotations and 22 validation annotations excluded are Categories 10 and 11, plus a small number of annotations released as `TBD`. Table 2 shows the category distribution in the training set.

5.2 Implementation Details

We train for 5,500 forward/backward iterations with gradient accumulation 2, corresponding to 2,750 optimizer updates and 275 linear warmup steps (10% of optimizer updates). Maximum sequence length is 512 tokens. Training time ranges from 38 minutes (CE variants) to 64 minutes (R-Drop) on a single NVIDIA T4 (16 GB VRAM). The full hyperparameter list is in Appendix A; compute and reproducibility details, including the train-

Cat	DICTUM Type	Spans	%
1	Contact related	2,293	5.28
2	Gathering additional information	376	0.87
3	Defining problem	17,015	39.19
4	Treatment goal	107	0.25
5	Drug related	10,930	25.17
6	Therapeutic procedure related	5,311	12.23
7	Evaluating test result	5,806	13.37
8	Deferment	84	0.19
9	Advice and precaution	1,496	3.45
	Total	**43,418**	**100.00**

Table 2: Category distribution in the official training split (350 documents, 43,418 in-scope spans across Categories 1–9). Categories 10 and 11 are present in the broader MedDec annotations but are out of scope for MedExACT and were excluded. Categories 2, 4, and 8 are extremely rare, together accounting for only 1.31% of all spans.

ing and inference cost of R-Drop and 4-view TTA, are in Appendix B. All experiments use a fixed random seed (42); no variance estimates are available, so small deltas and component gains should not be interpreted as statistically reliable. Validation scores reported in Table 3 use an internal proxy scorer that matches spans by normalized text rather than strict character offsets, and are not directly comparable to the official evaluator used in Table 5.

5.3 Baselines

We compare against the RoBERTa baseline from Elgaar et al. (2024), which achieves span F1 of 34.8 and token-level accuracy of 79.9 on the original MedDec test split (that evaluation uses a different split and scorer than MedExACT, so numbers are not directly comparable).

6 Results

6.1 Training Objective Comparison

Table 3 reports single-seed proxy validation scores for four training objectives. R-Drop + CE + soft-F1 produced the highest scores on all three metrics in this run. The CE + soft-F1 baseline reached Overall_F1 0.4996, 1.24 points below R-Drop, but this comparison lacks a seed sweep or confidence interval. Focal + soft-F1 matched the baseline on Base_Score (0.5441 vs. 0.5493) but fell further on Worst-Group F1 (0.4209 vs. 0.4499). Weighted CE + soft-F1 was the worst configuration across all metrics, with loss scale collapse at initialization (step-0 loss ≈ 0.04 vs. normal ≈ 2.9) destabilizing BIO training.

Training objective	Base	WG	Overall
CE + soft-F1	0.5493	0.4499	0.4996
Weighted CE + soft-F1	0.4652	0.3772	0.4212
Focal + soft-F1	0.5441	0.4209	0.4825
R-Drop + CE + soft-F1	**0.5570**	**0.4669**	**0.5120**

Table 3: Proxy validation scores for training objective comparison (0–1 scale). WG = Worst-Group F1. Overall = (Base + WG)/2. Scores use an internal proxy scorer; official test results are in Table 5. For reference, the RoBERTa baseline of Elgaar et al. (2024) achieves 34.8 span F1 on the original MedDec test split, which uses a different evaluation protocol and is not directly comparable.

Approach	Proxy Base	Δ
Baseline (BiomedBERT-ft, CE)	41.54	–
Focal loss (γ=2.0)	16.78	−24.8
Weighted CE (inv. freq.)	17.82	−23.7
Group DRO (η=0.01)	15.31	−26.2
BioLinkBERT-large (333M)	35.73	−5.8
Sliding window conv (w=3,5,7)	38.94	−2.6
Threshold calibration (per-cat)	31.67	−9.9

Table 4: Failed approaches tested during development. Proxy Base is reported on a 0–100 scale (i.e., percentage points), consistent with the baseline value of 41.54. Δ is the absolute difference from the CE baseline.

Table 4 lists additional configurations explored during development that did not improve over the CE baseline and were not carried forward.

6.2 Official Test Results

Table 5 shows official test set results. Our system placed fifth in the shared task with Overall_F1 of 0.5724. Token F1 of 0.6796 is the strongest among the top-five final-score submissions. We attribute this primarily to the soft token-F1 auxiliary loss, which optimizes the same token-level score that the official scorer reports. The result does not by itself validate the sparse routing design; routing and length calibration are better viewed as post-processing heuristics motivated by boundary and rarity analyses. The 19-point gap between Token F1 (0.6796) and Span F1 (0.4900) was consistent across all configurations and reflects the structural mismatch between token-level BIO labeling and exact character-span evaluation.

7 Analysis

7.1 Effect of R-Drop on Subgroup Robustness

The R-Drop term in Equation 4 accounts for 1.24 Overall_F1 points in the single-seed validation

Metric	Score
Span F1	0.4900
Token F1	0.6796
Base_Score	0.5848
Worst-Group F1	0.5601
Overall_F1	**0.5724**

Table 5: Official test set results. Base_Score = (Span F1 + Token F1)/2. Worst-Group subgroup is African American. Overall_F1 = (Base + WG)/2.

comparison (Table 3). The gain is larger on Worst-Group F1 (+1.70 points: 0.4499 to 0.4669) than on Base_Score (+0.77: 0.5493 to 0.5570). This is consistent with the intuition behind R-Drop (Liang et al., 2021): the KL consistency term reduces disagreement across dropout masks, which may help harder inputs where the model is less confident. Because only a single random seed (42) was used, statistical significance cannot be confirmed and the size of the R-Drop gain may be unstable.

7.2 Demographic Subgroup Performance

Table 6 and Figure 4 report per-subgroup test scores. Figure 4a shows a 10.1-point Base_Score gap between the best group (Hispanic, 0.6611) and the worst (African American, 0.5601). The African American subgroup has the lowest Span F1 (0.4639) and one of the lowest Token F1 scores (0.6562). Figure 4b also shows that the $\approx$ 19-point aggregate Token–Span gap holds broadly across subgroups (per-subgroup range 14.7–21.8 points), so the gap is primarily a system-wide property rather than one concentrated in the worst group. Structural analysis reveals that African American notes are harder: mean length 1,756 tokens vs. 1,570 for White notes, 140.3 annotations per note vs. 121.6, and a Cat7 share of 15.4% vs. 11.6% ($p = 0.0095$, Mann-Whitney). Readability (Flesch-Kincaid: $p = 0.50$) and abbreviation density ($p = 0.59$) did not differ significantly, ruling out lexical complexity as the primary driver.

7.3 Boundary Error Analysis

Figure 5 breaks down span matches per category into exact matches, near-misses (correct category but shifted offsets), and complete misses. Out of 7,022 in-scope gold spans in the official validation split, 3,373 (48.0%) are exact matches, 1,449 (20.6%) are near-misses, and 2,200 (31.3%) are completely missing. Appendix C reports the same data as per-category recall.

Subgroup	Span	Token	Base
Hispanic	0.5856	0.7366	0.6611
Female	0.5279	0.7149	0.6214
Other	0.5414	0.6879	0.6146
Asian	0.4800	0.6982	0.5891
English	0.4903	0.6872	0.5887
White	0.4812	0.6782	0.5797
Non-English	0.4893	0.6690	0.5792
Male	0.4677	0.6544	0.5611
Afr. Am.	**0.4639**	**0.6562**	**0.5601**

Table 6: Official test set scores per demographic subgroup, sorted by Base_Score descending. The African American subgroup is the Worst-Group bottleneck. The 10.1-point gap between the best (Hispanic) and worst group motivates future work on training-time fairness interventions.

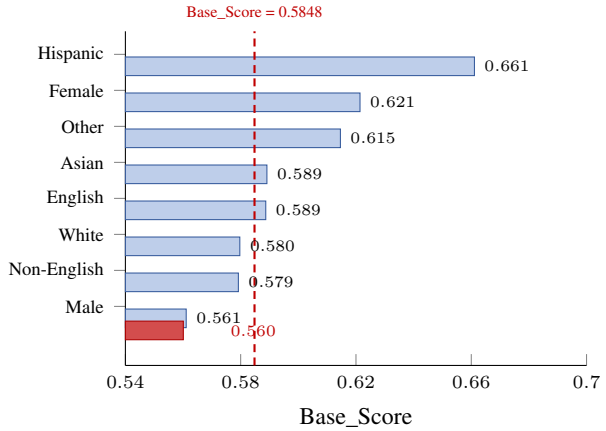
Left-boundary errors dominate: `pred_too_short_left` (642 instances) is the most frequent type, followed by `pred_too_long_left` (355). Figure 6 shows the full breakdown. Numbered bullets (*"1."*, *"2."*, *"3."*) are the most common missed left-boundary tokens – the model excludes list-item prefixes that are inside the gold annotation boundary.

7.4 Span-Length Distribution and Routing Motivation

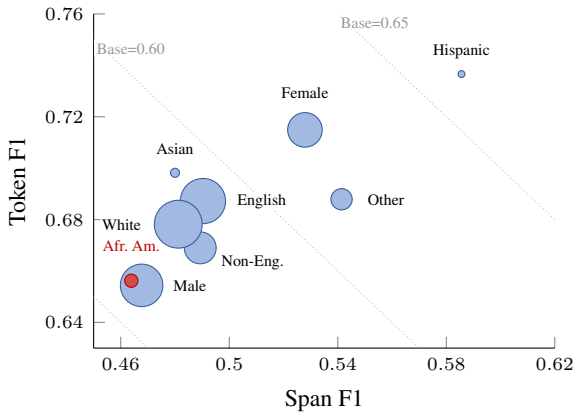
Figure 7 shows that Cat7 spans are much longer and more variable than other frequent categories. Cat4 does not show a significant length shift ($p = 0.21$) and is routed to the specialist branch only as a rare-category heuristic (107 training spans), where the weighted aggregation marginally recovers low-frequency spans in development runs. Cat9 (Advice and precaution) contains relatively long spans, motivating its 24-character minimum-length filter, which removes spurious short predictions.

7.5 Ablation Study

Table 7 reports single-seed incremental proxy validation scores as each component is added. These scores are *not* directly comparable to the official test results because the proxy scorer matches spans by normalized text content while the official scorer requires exact character offsets. R-Drop is the only component with a clear positive proxy delta in this run (+1.24 Overall, +1.70 WG), although this remains a single-seed observation. Adding TTA actually reduces proxy Base from 0.5570 to 0.5408, which at first looks counterproductive. The explanation is that TTA moves predicted span boundaries to better positions on average, which the official



(a) Base_Score by subgroup, sorted. The dashed line is the overall test-set Base_Score (0.5848); the red bar is **African American**, the worst group.



(b) Span F1 against Token F1 by subgroup, with marker area proportional to n. Dotted diagonals are lines of equal Base_Score. The $\approx$ 19-point aggregate Token–Span gap holds broadly across subgroups (per-subgroup range 14.7–21.8 points), so the gap is primarily a system-wide property rather than one concentrated in any one subgroup.

Figure 4: Test performance by subgroup. (a) Base_Score per group, sorted, with the overall test-set Base_Score shown as a dashed line and the worst group in red. (b) Span F1 against Token F1 for the same groups, with marker area proportional to group size and dotted diagonals marking equal Base_Score. Hispanic (n=89) scores highest and African American (n=313) sets the Worst-Group F1 at 0.5601.

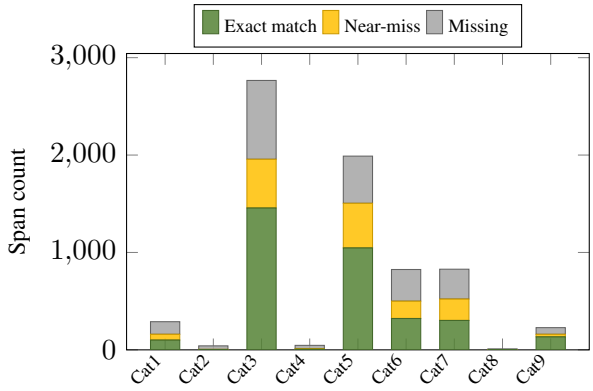


Figure 5: Per-category span match breakdown on the official validation split (53 documents; 7,022 in-scope gold spans across Categories 1–9). Near-misses (correct category, incorrect offsets) account for 20.6% of all gold spans, confirming boundary recovery as the dominant error mode. Categories 2, 4, and 8 are nearly invisible to the model due to extreme rarity.

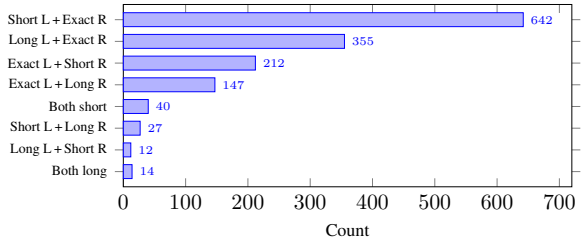


Figure 6: Directional breakdown of 1,449 boundary errors. Left-boundary errors (short or long on the left) account for 71.4% of near-misses. L = left boundary, R = right boundary.

character-offset scorer rewards but the proxy text-match scorer penalizes when the normalized content was already correct. TTA appeared beneficial under the official scorer, but a clean controlled final-system ablation was not recoverable from the saved artifacts. Length calibration shows no change on the proxy scorer because the constraints act on character lengths, which the text-match scorer ignores; its value is measured by the official evaluator. The final sparse-routing step changes Overall_F1 by only +0.0010 relative to length calibration on the proxy scorer, so it should not be interpreted as a robust validation gain.

7.6 Qualitative Error Analysis

The two categories routed to the Specialist Branch exhibit distinct failure patterns. Cat7 (Evaluating test result) spans are long (mean 79.9 chars) and often straddle chunk boundaries; a span like *"CXR showing bilateral infiltrates, unchanged from prior"* is frequently truncated to the first clause when a chunk boundary falls mid-sentence. The half-offset TTA and context-heavier Specialist Branch were designed to reduce this fragmentation. Cat4 (Treatment goal) is extremely rare (107 training spans); the model routinely predicts zero Cat4 spans on

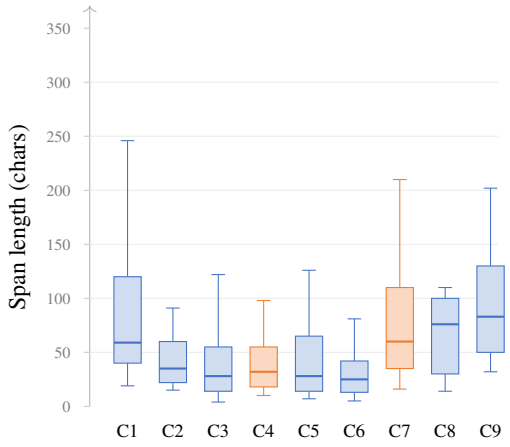


Figure 7: Gold span-length distributions per category (character count). Orange: categories routed to the Specialist Branch (C4, C7). Blue: Anchor categories. C7 has the widest IQR and highest median among frequent categories ($n > 40$).

Configuration	Base	WG	Overall
CE + soft-F1 only	0.5493	0.4499	0.4996
+ R-Drop (α=0.5)	0.5570	0.4669	0.5120
+ Viterbi decoding	0.5570	0.4669	0.5120
+ 4-view TTA	0.5408	0.4640	0.5024
+ Length calibration	0.5408	0.4640	0.5024
+ Sparse routing (Cat 4, 7)	**0.5407**	**0.4660**	**0.5034**

Table 7: Incremental ablation (proxy scorer). TTA and length calibration appear neutral or negative on this scorer but were retained in the final official system: the proxy matches normalized text content and cannot capture exact-offset gains from TTA or length trimming. The sparse-routing delta is very small and is not treated as evidence of a stable improvement.

many notes, and the Specialist Branch provides a distributional shift that marginally recovers some of these low-frequency spans, but this choice is not supported by the length test.

We also observe clinically plausible false positives, suggesting that some errors reflect mismatch between model behavior and annotation guidelines rather than simple confidence miscalibration.

8 Conclusion

We have presented our MedExACT 2026 system for extracting nine-category medical decision spans from ICU discharge summaries. The system uses a single BiomedBERT checkpoint with no extra parameters at inference.

R-Drop was the strongest training decision in our single-seed validation comparison, adding 1.24 Overall_F1 points with a larger 1.70-point gain on Worst-Group F1. The larger gain on WG than on Base suggests that the KL consistency term may help harder or underrepresented inputs where the model is least confident, but the absence of a seed sweep prevents a stability claim.

Four-view TTA and sparse category routing did not improve proxy validation scores, which are based on text-match rather than offset accuracy. These components are part of the best official system and are consistent with the observed boundary and long-span error patterns, although clean controlled final-system ablations were not recoverable. Within sparse routing, Cat7 has strong length-based support; Cat4 routing is an exploratory rare-category heuristic and should not be treated as statistically justified by span length.

The persistent 19-point gap between Token F1 (0.6796) and Span F1 (0.4900) shows that the model largely finds the right text but consistently gets the exact start and end characters wrong. This is a fundamental limitation of BIO token tagging, which cannot model overlap and has no direct loss on character offsets. Moving to span-boundary or span-classification architectures that directly optimize character-level match criteria is the natural next step.

Limitations

Single seed. All training runs use seed 42; no variance estimates are available and all reported gains should be interpreted as single-run observations. This particularly affects the reported R-Drop gain and the very small sparse-routing delta. A complete evaluation would report means and standard deviations across multiple seeds or confidence intervals from resampling.

Proxy vs. official scorer. Validation scores in Tables 3 and 7 use an internal proxy scorer that matches spans by normalized text rather than strict character offsets. The proxy scorer and the official shared-task evaluator differ in token-level scoring behavior, so validation and test numbers are not directly comparable.

Sparse routing evidence. Cat7 routing is supported by a significant span-length difference and a high near-miss rate, but Cat4 routing is not supported by a significant length difference ($p = 0.21$). Cat4 was included because it is rare and sometimes benefits from a context-weighted branch in development, so this part of the routing recipe should be viewed as heuristic. The Specialist Branch weights

$[0.4, 0.4, 0.1, 0.1]$ were chosen to favor broader context windows but were not selected by a formal grid search.

Fairness-aware optimization scope. Our use of "fairness-aware optimization" should be interpreted as metric-aligned development rather than an effective training-time fairness method. The system's fairness contribution is selection-time alignment by Overall_F1, not training-time robustness. Group DRO was explored and failed in this formulation. Dedicated subgroup training, lexical error analysis by subgroup, variance estimates, and subgroup-targeted significance testing remain as future work.

Ethics Statement

This work uses the MedDec dataset derived from MIMIC-III, which contains de-identified clinical notes from the Beth Israel Deaconess Medical Center. Access requires completion of a data use agreement and CITI training through PhysioNet. Our system processes only de-identified text and does not attempt re-identification. The fairness-aware evaluation metric used in MedExACT penalizes models that perform unevenly across demographic subgroups, and we report per-subgroup scores in full (Table 6). We note that our system addresses fairness only through model selection, not through training-time interventions, and the 10-point gap between the best and worst subgroups remains an open problem.

Acknowledgments

We thank the MedExACT organizers for providing the MedDec dataset and evaluation infrastructure.

References

Siyu Ding, Junyuan Shang, Shuohuan Wang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2021. ERNIE-Doc: A retrospective long-document modeling transformer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2914–2927, Online. Association for Computational Linguistics.

Mohamed Elgaar, Hadi Amiri, Mitra Mohtarami, and Leo Anthony Celi. 2025. MedDecXtract: A clinician-support system for extracting, visualizing, and annotating medical decisions in clinical narratives. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 481–489, Vienna, Austria. Association for Computational Linguistics.

Mohamed Elgaar, Jiali Cheng, Nidhi Vakil, Hadi Amiri, and Leo Anthony Celi. 2024. MedDec: A dataset for extracting medical decisions from discharge summaries. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 16442–16455, Bangkok, Thailand. Association for Computational Linguistics.

Mohamed Elgaar, Jiali Cheng, Nidhi Vakil, Mehrnaz Sadrolashrafi, Mitra Mohtarami, Adrian Wong, Hadi Amiri, and Leo A. Celi. 2026. Overview of medical decision extraction, analysis, and classification task (medexact) 2026. In *The 25th Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, San Diego, California, USA. Association for Computational Linguistics. To appear.

Jinlan Fu, Xuanjing Huang, and Pengfei Liu. 2021. SpanNER: Named entity re-/recognition as span prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7183–7195, Online. Association for Computational Linguistics.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3:160035.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2021. A span-based model for joint overlapped and discontinuous named entity recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4814–4828, Online. Association for Computational Linguistics.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-Drop: Regularized dropout for neural networks. In *Advances in Neural Information Processing Systems*, volume 34.

Evan Zheran Liu, Behzad Haghgoo, Annie S. Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. 2021. Just train twice: Improving group robustness without training group information. In *Proceedings of the 38th International*

Conference on Machine Learning, pages 6781–6792. PMLR.

Eirik H. Ofstad, Jan C. Frich, Edvin Schei, Richard M. Frankel, and Pål Gulbrandsen. 2016. What is a medical decision? A taxonomy based on physician statements in hospital encounters: A qualitative study. *BMJ Open*, 6(2):e010098.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *Proceedings of the International Conference on Learning Representations*.

Enwei Zhu and Jinpeng Li. 2022. Boundary smoothing for named entity recognition. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7096–7108, Dublin, Ireland. Association for Computational Linguistics.

Appendices

A Hyperparameter Summary

Table 8 and Table 9 summarize the hyperparameters used in our final submitted system.

Training	
Training steps	5,500
Batch size	8 (eff. 16 w/ grad. accum. 2)
Learning rate	2×10^{-5}
Warmup steps	275 (10% of 2,750 optimizer updates)
Optimizer	AdamW
Weight decay	0.01
Epsilon	10^{-8}
Gradient clip	1.0 (max norm)
Max sequence length	512 tokens
Label smoothing ϵ	0.10
Soft-F1 weight λ	0.20
R-Drop weight α	0.50
Positive-window bias	0.70
Random seed	42
Precision	fp16 (autocast / GradScaler)

Table 8: Training hyperparameters for the final submitted system.

Inference & Post-processing	
TTA configurations	$(384, 0)$, $(384, 192)$; $(256, 0)$, $(256, 128)$
Specialist weights	[0.4, 0.4, 0.1, 0.1] (logit space)
Cat. 2 max span length	98 chars
Cat. 6 min span length	3 chars
Cat. 9 min span length	24 chars
Cats. 1,3,4,5,7,8	unconstrained
Viterbi start state	O or any B-$*$
I-k allowed after	B-k or I-k only

Table 9: Inference, post-processing, and decoding constraints.

B Compute and Reproducibility

Hardware. All training and inference was run on an environment with two NVIDIA T4 GPUs (16 GB VRAM per card, 32 GB total), using fp16 autocast with `GradScaler`. The BiomedBERT-base encoder has approximately 110 M parameters; with the additional 19-way linear BIO classification head the full model fits comfortably on a single T4. Individual training runs used a single T4; the second card was reserved for parallel ablation sweeps.

Training cost. Single-pass training runs (every configuration in Table 3 except R-Drop + CE + soft-F1) complete the 5,500 forward/backward iterations in approximately 38 minutes. R-Drop adds a second forward pass per training step; the R-Drop + CE + soft-F1 configuration takes about 64 minutes. Total wall-clock time across the four training-objective comparison runs in Table 3 is under 3 hours on one T4.

Inference cost. Single-view inference (one sliding window with default stride) is the baseline cost unit. The 4-view TTA ensemble passes each document through the encoder with the four window configurations in Table 9, so inference compute scales roughly $4\times$ over single-view. The sparse routing step adds no extra model cost because both branches read from the same checkpoint; length calibration and deduplication are pure Python post-processing.

C Per-Category Validation Breakdown

Figure 5 reports per-category match counts on the 53-document validation split. Table 10 expresses those counts as exact and near-miss recall. Three patterns are visible.

Recall tracks training-set frequency. Categories 2, 4, and 8 have 376, 107, and 84 training spans respectively (Table 2), and reach exact recall of at most 0.152 on validation. Cat 3 alone has 17,015 training spans, more than 200 times Cat 8, and its exact recall is 0.527. A BIO classifier trained on this split cannot close that frequency gap without auxiliary supervision, so the rare categories contribute little to aggregate metrics.

Cat 7 fails at boundaries. Its exact recall (0.365) is 2.5 percentage points below Cat 6 (0.390), but its near-miss rate of 26.6% is the highest of any category. This is consistent with its long span-length distribution (median 60 characters, Figure 7): long

spans are more likely to have boundary errors under the strict span matcher. This is the empirical motivation for routing Cat 7 through the weighted Specialist Branch (Section 4.5).

Cat 9 benefits from length calibration. Its exact recall (0.588) is the highest of any category despite its mid-range training frequency (1,496 spans). The 24-character minimum-length filter (Section 4.6) discards short noisy predictions that would otherwise be scored as false positives under the strict matcher, which improves precision on Cat 9; the threshold was chosen on validation to avoid deleting short but valid spans, so recall is preserved.

	Counts			**Recall**	
Cat	Exact	Near	Miss	Exact	Exact+near
1	101	59	129	0.349	0.554
2	2	3	36	0.049	0.122
3	1,457	500	809	0.527	0.708
4	7	3	36	0.152	0.217
5	1,047	458	484	0.526	0.757
6	322	178	325	0.390	0.606
7	302	220	306	0.365	0.630
8	1	2	7	0.100	0.300
9	134	26	68	0.588	0.702

Table 10: Per-category validation-split recall, derived from the counts in Figure 5. **Exact** recall treats only strict matches as correct. **Exact+near** counts any predicted span that carries the correct category, regardless of offset. Gold spans per category are the row sums of the *Counts* block.