Segmentation Matters: Exploring LLM-Based Strategies for Temporal Clinical Event Identification in Oncology Reports

Cristiano Bellucci[4], Francesco Madeddu[2,3], Chiara Iacomini[1], Carlotta Masciocchi[1], Stefano Patarnello[1], Massimo Bernaschi[2], Mario Santoro [2], Livia Lilli [1]

[1] Fondazione Policlinico Universitario Agostino Gemelli IRCCS, Rome, Italy

[2] Istituto per le Applicazioni del Calcolo "Mauro Picone", Italian National Research Council

[3] Department of Computer, Control and Management Engineering, Sapienza University of Rome

[4] KeyBiz srl

livia.lilli@policlinicogemelli.it

Abstract

Processing unstructured clinical narratives remains a major challenge in medical Natural Language Processing (NLP), particularly when critical information is embedded within lengthy and heterogeneous reports. Clinical notes often describe key diagnostic and therapeutic events through a verbose narrative, making automatic event identification difficult. In this work, we frame the identification of clinical events as a text segmentation task. We conduct a comparative study of three segmentation strategies applied to oncology reports: (i) a fully regex-based approach, (ii) a cascaded regex–LLM pipeline, and (iii) the same cascade architecture augmented with a recovery mechanism to mitigate LLM rephrasing. Segmentation quality is evaluated using complementary structural metrics (Pk, WindowDiff, Boundary Similarity, Segment Count Accuracy, and Text Overlap IoU), and its impact is also observed on downstream segment tagging, performed to identify the corresponding event type (e.g. surgery, biopsy, imaging, treatment, laboratory). The results demonstrate the high potential of LLM-based approaches, particularly in preserving semantic coherence within segments and generalization on new data sources. However, regex-based segmentation achieves higher performance according to structural segmentation metrics, also leading to better downstream clinical event identification. In general, these results highlight the critical role of context-adaptive high-quality segmentation strategies in the structuring of verbose clinical narratives and in the accurate identification of key patient events.

1 Introduction

Processing unstructured clinical narratives remains a major challenge in medical Natural Language Processing (NLP). Clinical reports are typically lengthy and unstructured documents that aggregate heterogeneous information in a verbose narrative, including key events like diagnostic findings, therapeutic interventions, laboratory results, and follow-up plans. This structural complexity makes them difficult to handle within downstream NLP pipelines, where models often require well-defined and coherent input units.

To address this issue, preprocessing strategies are commonly employed to make clinical texts more workable. In particular, text segmentation can support downstream tasks by dividing reports into smaller and semantically coherent units, facilitating more targeted processing. However, chunking techniques used in transformer-based and LLM architectures are predominantly length-driven rather than meaning-driven (Jaiswal and Milios, 2023). As a consequence, they may split text at arbitrary boundaries, disrupting semantic coherence and potentially separating information that belongs to the same clinical event, or they may fail to identify semantically distinct sub-sections, relative to two different events. In this direction, Lilli et al. (2026) showed that content-aware segmentation of clinical reports can improve downstream information extraction by enabling content-specific prompting strategies for different clinical event types. In that work, the primary focus was not the segmentation task itself, but the evaluation of extraction performance under different prompt configurations tailored to the segmented report content. Within this framework, segmentation represented a key preprocessing step for the extraction architecture and was implemented through a preliminary regex-based approach.

In this study, we reframe text segmentation as a semantics-driven task aimed at identifying clinically meaningful event units, and not only reducing input length. Focusing on the clinical-history section of oncological reports, we seek to automatically detect and isolate key temporal events (such as imaging examinations, surgeries, laboratory assessments, biopsies, and treatment lines) that define the patient's clinical trajectory. These

Proceedings of the 25th Workshop on Biomedical Language Processing (BioNLP 2026), pages 595–604

events are typically embedded within chronological narrative text, where explicit structural markers may be inconsistent or absent, making boundary detection particularly challenging. To address this problem, we conduct a comparative evaluation of three segmentation strategies: (i) a fully rule-based approach relying on regular expressions and structural cues, (ii) a regex–LLM pipeline that refines rule-based segments through semantic modeling, and (iii) the hybrid regex-LLM framework incorporating a recovery mechanism to mitigate LLM-induced boundary errors and hallucinations. In addition to intrinsic segmentation metrics, we evaluate each strategy based on its downstream event tagging performance, demonstrating that segmentation quality directly affects the accuracy of event type classification. Through this analysis, we highlight the complementary strengths of deterministic rules and LLM-based approaches, and demonstrate the importance of high-quality text segmentation for accurately identifying key events within a patient's clinical trajectory. The study framework is shown in Figure 1.

2 Background

Early approaches to document structuring relied heavily on rule-based systems to identify section boundaries. Pioneering works like SecTag (Denny et al., 2009) utilized terminological headers and line formatting to detect document sections, a methodology that remains influential in modern pipelines such as medspaCy (Eyre et al., 2021). Although these approaches effectively handle document structures, they often lack the granularity required to tell apart distinct clinical events within narrative-heavy sections like the clinical history, where explicit headers are frequently replaced by chronological descriptions and date references.

The granular identification of such events is closely linked to temporal information extraction. Major initiatives like the i2b2 Shared Task (Sun et al., 2013) and the THYME corpus proposed for use in a SemEval 2015 task (Styler IV et al., 2014) established the standard for extracting events, temporal expressions, and their relationships. These tasks were approached mostly with ML supervised methods, which typically require large, annotated corpora (scarce in real-world settings).

Furthermore, recent work by Lilli et al. (2025b) shows how sentence boundary detection helps text segmentation using both traditional and LLM-based methods, including SaT (Frohmann et al., 2024), PySBD (Sadvilkar and Neumann, 2020), Stanza (Qi et al., 2020) and NLTK (Bird, 2006). The benchmark analysis shows that even sentence-level boundary detection can significantly enhance downstream clinical report classification and critical event identification.

Building on the above findings, text segmentation emerges as a crucial and actively explored task in clinical NLP. However, despite its importance, it remains challenging due to the variability and complexity of clinical narratives. With recent advances in NLP, many clinical text processing tasks are increasingly being addressed using LLM-based approaches, which have enabled substantial progress in areas such as clinical information extraction and medical text generation (Yang et al., 2022; Agrawal et al., 2022; Lilli et al., 2025a). Recent comparative studies, such as CNSight (Surana et al., 2025), have evaluated rule-based, transformer, and LLM-based approaches for clinical note segmentation, similarly finding that rule-based methods can be robust. While with CLINES (Yang et al., 2025b), the complex space of clinical text segmentation for semantical analysis is further explored.

Despite their strong semantic understanding, LLMs remain prone to hallucinations, often producing plausible yet clinically inaccurate information (Ji et al., 2023). To mitigate this limitation, hybrid approaches have been proposed that combine the deterministic precision of regex with the reasoning capabilities of LLMs (Kleinlein et al., 2025). Crucially, a major challenge in LLM-based segmentation is ensuring that the generated output aligns exactly with the legally valid source text. Recent efforts such as LangExtract (Goel, 2026) and SafePassage (Barrow et al., 2025) focus heavily on grounding LLM outputs back to source spans. The closest work to our proposed recovery approach is MedSlice (Davis et al., 2025), which addresses LLM boundary misalignment in section-level segmentation using Levenshtein-based fuzzy matching. Building upon this line of research, our work introduces an extended recovery mechanism that leverages full-segment fuzzy matching rather than anchor n-grams, incorporates gap-filling between aligned segments, and implements a global rollback strategy when alignment fails.

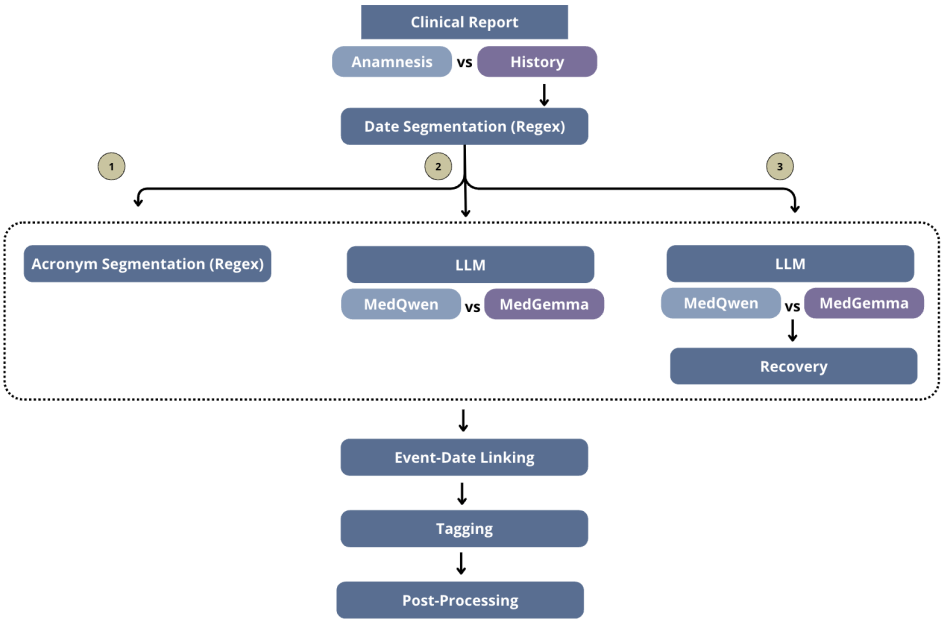

Figure 1: Study workflow with the 3 segmentation strategies: (1) regex-only, (2) regex + LLM, (3) regex + LLM + recovery.

3 Method

3.1 Dataset

The dataset consists of anonymized, real-world Italian ovarian oncology reports from multidisciplinary tumor board sessions at the Gemelli Hospital in Rome, where clinicians discuss patient cases and decide on follow-up and care. The documents are typically extensive and semi-structured. Each report opens with an anamnestic summary of the patient's current status, followed by a clinical history section. This history is organized as a sequence of events (such as surgeries, diagnostic imaging, or systemic treatments) which are generally prefaced by an explicit date or a functional header that provides a brief summary of the event type (e.g., *TC total body* for an imaging event). The primary objective of this study is to leverage this structure to automatically isolate these clinical events, transforming the narrative history into independent textual segments.

3.2 Text Segmentation

The study considers three text-segmentation strategies applied exclusively to the clinical history section of the reports: (1) a regex-only method, (2) a cascaded regex–LLM pipeline, and (3) the same cascaded pipeline augmented with a recovery step.

3.2.1 Regex-only

The first approach identifies event boundaries by detecting date expressions strictly at the beginning of a sentence or line-initial position. Supported date formats include `dd/mm/yyyy`, `dd/mm/yy`, month–year expressions (e.g., `month, year`), and `dd/mm` without year. Dates preceded solely by plain text are ignored to avoid spurious intra-sentential matches.

After date-based segmentation, further refinement is performed by analyzing patterns that may correspond to event titles or section headers. This step is designed to capture clinical events that lack an explicit date but are introduced by a functional header. To this purpose, a domain-specific dictionary of acronyms and lexical variants used to denote clinical events is employed to support boundary detection. Table 1 reports the Italian patterns used to identify event types. We focus on the key clinical events in the oncological patient trajectory, namely: imaging (including instrumental examinations such as CT, MRI, X-ray, and ultrasound), surgery, laboratory tests (where biomarker values are typically reported), treatment, and biopsy. Finally, flow 1 in Figure 1 represents the Regex-only segmentation.

3.2.2 Regex + LLM

In the two-stage cascade approach, the clinical-history section is first segmented into event units

using the date-based regex rules described above. Each resulting segment is then provided as input to an LLM, to which it is explicitly requested to identify further sub-segments corresponding to key clinical events, based on a predefined list of domain-specific acronyms. These acronyms are used as anchors to detect meaningful intra-segment boundaries that typically introduce events not detected through the above date-regex approach. The LLM returns a structured JSON output, which is automatically parsed to extract the corresponding facts and store them in a normalized representation for a granular segmentation. Flow 2 in Figure 1 represents this strategy. The prompt used is the following (we report the English version for better understanding):

```
Segment the following clinical text into separated
facts based ONLY on medical acronyms.

ACRONYMS that indicate a new segment:
- Imaging: {list of acronyms}
- Surgery: {list of acronyms}
- Laboratory: {list of acronyms}
- Therapy: {list of acronyms}
- Biopsy: {list of acronyms}
- Other: {list of acronyms}

Provide the output ONLY in JSON format without
preambles or explanations.
The facts must correspond exactly to the segments
     of
the clinical report.
Example output: {fact1: <text of fact 1>, ..}.

If there are no acronyms, return a JSON with a
single fact: {fact1: <text of fact 1>}.

TEXT: {input text}
```

Event type	Acronyms / keywords
Imaging	TC; TAC; TCTAP; TCTB; RM; RMN; PET; ECO; Ecografia; ETG; ECOTV; RX
Surgery	Laparoscopia; LPS; Laparotomia; LPT; Intervento chirurgico
Laboratory	CA-125; CA125; CEA; HE4; AFP; Markers; Marcatori
Therapy	NACT; Chemio; CHT; I LINEA; II LINEA; III LINEA
Biopsy / Histology	Biopsia; Istologico; EID

Table 1: Domain-specific acronyms and lexical variants used for event segmentation and tagging. All reported acronyms and keywords correspond to synonyms or abbreviations of the respective event type.

3.2.3 Regex + LLM + Recovery

Building on recent grounding techniques such as those proposed in MedSlice (Davis et al., 2025), this third strategy extends the previous cascaded regex+LLM pipeline by introducing an additional recovery procedure. This was necessary because, despite explicit instructions to extract segments verbatim, LLMs frequently introduce subtle rephrasings, summarize content, or omit punctuation. These "hallucinations of form" prevent direct string comparison and necessitate fuzzy alignment to map the LLM's semantic boundaries back to the original, legally-valid source text. After the initial regex-based date segmentation and the subsequent LLM refinement, this module verifies and reconstructs segments through the following steps:

1. **Fuzzy Matching.** Each LLM-generated segment is aligned to the original document using fuzzy string matching with the available Python library[1], which returns a similarity score s and the corresponding character offsets. Defining a threshold τ, if $s \geq \tau$, the segment is considered valid and the exact substring is extracted from the original text using the matched positions.

2. **Gap Filling.** Segments with $s < \tau$ are marked as invalid. When an invalid segment is located between two valid segments, it is replaced by directly recovering the original text spanning the gap between the end of the preceding valid segment and the start of the following one, ensuring complete coverage of the source document.

3. **Rollback.** If the proportion of the remaining invalid segments exceeds a predefined threshold ρ, the LLM-based segmentation is discarded entirely and the system reverts to the regex-only method.

Flow 3 in Figure 1 represents this approach. After a preliminary grid search over the alignment parameters, we selected the configuration that provided the best balance between alignment precision and coverage. Specifically, we set the similarity threshold to $\tau = 85\%$ and the Rollback ratio to $\rho = 0.5$, as this combination yielded the most stable and consistent performance across our experiments.

3.3 Event-Date Linking

After segmentation using one of the three previously described strategies, we introduce an additional processing step aimed at associating a temporal reference with each extracted event before performing event tagging.

[1] https://github.com/rapidfuzz/RapidFuzz

For segments obtained through date-based separation, the associated date is directly inherited from the date expression used as the segmentation boundary.

In contrast, for segments generated through the secondary step (which is acronym-based rules in the fully-regex strategy or LLM-based segmentation in the cascade approaches), the event remains temporarily without an associated date and subsequently handled during the post-processing phase.

3.4 Event Tagging

The date-linked segments are assigned a single event label through a pattern-matching procedure. Specifically, each segment is scanned for the presence of the acronyms listed in Table 1, which define the event categories. When multiple patterns are present within the same segment, the label corresponding to the earliest occurring pattern is selected. This strategy ensures a deterministic and consistent tagging process for identifying the key diagnostic events in the patient trajectory.

3.5 Post-Processing

After tagging, the segments undergo a cleaning and normalization procedure to ensure temporal consistency and structural coherence. First, dates are normalized by expanding two-digit years using a pivot rule (years ≤ 25 mapped to the 2000s and ≥ 26 to the 1900s) and assigning the first day of the month to month–year expressions. Segments without an explicit date inherit the temporal reference from the immediately preceding segment. When only month and year information is available and the preceding segment specifies a full date within the same month, the current date is shifted to one day after the previous event (this imputation rule was agreed upon with the clinical experts to maintain the chronological sequence of the patient's history). Malformed or truncated years are corrected by locating the nearest valid four-digit year and inferring the appropriate value based on month comparison. Finally, consecutive segments with the same event label and identical date are merged through text concatenation. This post-processing step mitigates over-segmentation errors, ensuring that a clinical event incorrectly divided into multiple fragments, is reconstructed as a single, semantically coherent unit.

4 Experiments

4.1 Input Data

The original dataset comprises approximately 1600 patients with more than 5000 clinical notes written in Italian language. From this larger corpus, we constructed a gold standard consisting of 343 text segments for evaluation, derived from reports belonging to 20 patients. During the gold standard annotation process, a text segment was strictly defined as an independent temporal event. The segments and their associated event tags were manually annotated by a clinical expert. Furthermore, to ensure an unbiased evaluation, all regex patterns and the acronym dictionary were initially designed and tuned on a separate development set of reports that do not overlap with the 20 patient reports used for testing.

An overview of the gold dataset is provided in Table 2, while Figure 2 illustrates the distribution of event types across reports.

Characteristic	Value
Number of Reports	20
Total Segments	343
Mean Segments per Patient	17.1

Table 2: Gold Standard dataset statistics.

4.2 Segmentation Setup

All experiments were conducted on a workstation equipped with an NVIDIA L40S-48C GPU (48 GB VRAM). For the hybrid segmentation strategies, LLM inference was executed using `llama.cpp`[2]. We evaluated two domain-specific medical language models: MedGemma (27B) (Sellergren et al., 2025) and MedicalQwen3 (14B) (Yang et al., 2025a), in particular we used `unsloth/medgemma-27b-it-GGUF` and `mradermacher/MedicalQwen3-Reasoning-14B-IT-i1-GGUF`. Both models were deployed in GGUF format using `Q6_K` quantization. Furthermore, evaluation on segmentation was assessed through the comparison among the gold and computed segments, using five complementary metrics. Table 3 reports such results averaged over segments for each experiment, while further details about the metrics are listed below:

Pk. Measures the probability that two points k tokens apart are incorrectly classified as belonging

[2]https://github.com/ggml-org/llama.cpp

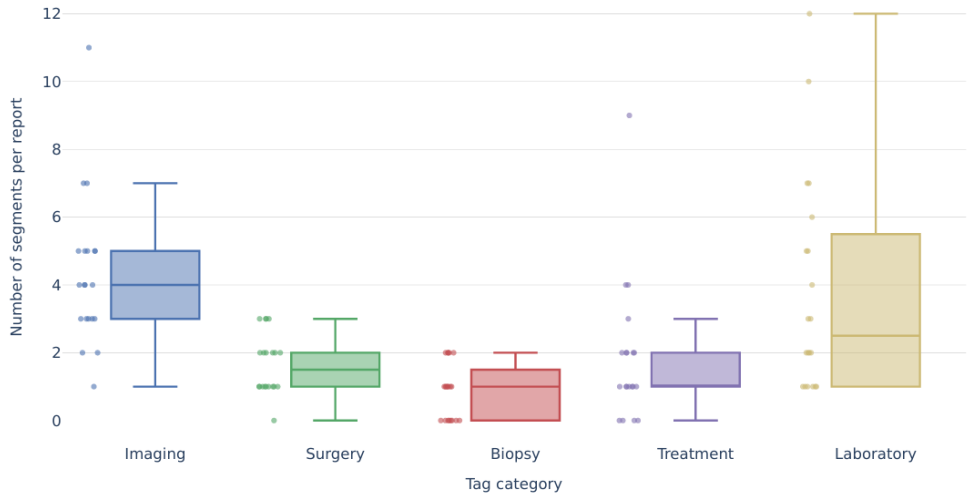

Figure 2: Distribution of segments per report, by tag event type.

to the same or different segments (lower is better) (Beeferman et al., 1999). For this metric, the pk function of the `SegEval` Python library was used.

WindowDiff. Counts boundary mismatches within a sliding window of size k, penalizing both missed and spurious boundaries (lower is better) (Pevzner and Hearst, 2002). For this metric, the `window_diff` function of the `SegEval` Python library was used (Fournier, 2013a).

Boundary Similarity. Computes an F1-style agreement between output and gold boundary positions (higher is better) (Fournier, 2013b). For this metric, the `boundary_similarity` function of the `SegEval` Python library was used (Fournier, 2013a).

Segment Count Accuracy. Quantifies structural agreement as $1 - |n_{\text{pred}} - n_{\text{gold}}|/n_{\text{gold}}$ (higher is better).

Text Overlap IoU. Measures the average Jaccard similarity between matched output and gold segments (higher is better).

Boundary-based metrics (Pk, WindowDiff, and Boundary Similarity) were computed at the *character level*. Predicted segments were converted into character-count ("mass") representations using the `SegEval` library (Fournier, 2013a). To account for minor formatting discrepancies introduced by LLMs (e.g., extra whitespaces), predicted masses were proportionally aligned to the gold-standard length through an `align_masses` procedure. For Pk and WindowDiff, the window size k was dynamically set to half the average gold segment length for each report. Conversely, Text Overlap IoU was computed at the *word level* using whitespace-tokenized segments and Jaccard similarity between greedily matched predicted and gold segments. Segment Count Accuracy and downstream F1 scores were computed globally across all reports.

4.3 Tagging Setup

Event classification of text segments was evaluated by comparing the gold and output segment tags using the F1 metric. The comparison relies on aligning gold-standard and predicted segments, which are not always perfectly matched: the number of output segments may differ from the gold standard, leading to over-segmentation or under-segmentation.

To address this, a fuzzy matching strategy was adopted. For each gold segment, the most semantically similar output segment is identified based on textual similarity. An output segment is considered a valid match only if its similarity with the corresponding gold segment exceeds an 80% threshold. This threshold is intentionally less strict than the one used in the recovery phase, as the goal here is to associate segments referring to the same underlying event type and enable a fair comparison during tagging evaluation, rather than to enforce precise boundary reconstruction.

Once matched, an output segment cannot be assigned to another gold segment. Unmatched gold segments (due to under-segmentation) are treated as False Negatives, while unmatched output seg-

Metric	Regex	Regex + MQ + Recovery	Regex + MG + Recovery	Regex + MQ	Regex + MG
Pk (↓)	**0.058 ± 0.080**	0.115 ± 0.094	0.076 ± 0.088	0.209 ± 0.125	0.149 ± 0.109
WindowDiff (↓)	**0.102 ± 0.096**	0.185 ± 0.110	0.125 ± 0.100	0.294 ± 0.139	0.225 ± 0.116
Boundary Sim. (↑)	**0.538 ± 0.353**	0.268 ± 0.265	0.459 ± 0.305	0.192 ± 0.249	0.288 ± 0.256
Seg. Count Acc. (↑)	**0.919 ± 0.091**	0.785 ± 0.209	0.855 ± 0.165	0.588 ± 0.311	0.679 ± 0.253
Text Overlap IoU (↑)	0.739 ± 0.248	**0.833 ± 0.127**	0.820 ± 0.196	0.806 ± 0.111	0.825 ± 0.149

Table 3: Segmentation metrics (mean ± std, $n = 20$) for the five setups. MQ = MedicalQwen3; MG = MedGemma. Arrows indicate whether lower (↓) or higher (↑) is better.

ments (due to over-segmentation or hallucinated boundaries) are treated as False Positives, ensuring that the reported F1 scores penalize segmentation errors.

Table 4 reports the results for each event type and overall across all experiments.

4.4 Results and Discussion

4.4.1 Segmentation Findings

Table 3 reports the segmentation performance across five configurations, where metrics are computed at the segment level and then averaged (mean ± std). Overall, the pure Regex strategy achieves the best results on most boundary-sensitive metrics, with the lowest Pk (0.058) and WindowDiff (0.102), and the highest Boundary Similarity (0.538) and Segment Count Accuracy (0.919), confirming the effectiveness of deterministic rules for precise boundary detection and segment preservation.

Among LLM-based approaches, MedGemma with Recovery shows the closest performance to Regex on structural metrics (Pk = 0.076; WindowDiff = 0.125; Seg. Count Acc. = 0.855), confirming the utility of Recovery in reducing boundary errors. Removing the Recovery mechanism leads to a clear degradation for both models (e.g., MedGemma Pk increases from 0.076 to 0.149), highlighting its stabilizing effect.

Interestingly, the best Text Overlap IoU is achieved by MedicalQwen3 with Recovery (0.833), slightly outperforming Regex (0.739). This suggests that, although LLM-based segmentation may not perfectly align with gold-standard boundaries, it captures semantic content.

The relatively low Boundary Similarity scores for LLM-based methods can be attributed to over- and undersegmentation effects. In particular, the introduction of spurious boundaries (oversegmentation) and occasional missed splits (undersegmentation) strongly penalize this F1-style boundary metric, even when Pk and WindowDiff indicate overall reasonable segmentation quality.

Finally, the relatively high standard deviations (especially for Boundary Similarity and Segment Count Accuracy) indicate notable inter-patient variability, likely due to differences in report structure and event density, which affect segmentation difficulty. Overall, Boundary Similarity is the most affected metric, showing high variability across all configurations, whereas Pk and WindowDiff remain more stable. The Regex approach exhibits the lowest standard deviations, likely because it was specifically engineered to handle the structural complexity of this dataset. However, when moving to different data sources, LLM-based techniques may offer greater generalizability.

4.4.2 Tagging Findings

Table 4 reports the F1 scores for event tagging under different segmentation strategies, both per category and overall. The results clearly show that segmentation quality directly impacts classification performance. In line with Table 3, the fully Regex-based segmentation (being the most accurate in boundary detection) also achieves the best overall F1 score (0.757), confirming that more precise segmentation leads to better downstream event classification.

At the category level, Regex outperforms the other methods in most cases (Imaging, Surgery, and Biopsy). However, the cascaded approaches with Recovery show competitive or superior results for Treatment (Regex + MedicalQwen3 + Recovery, 0.721) and Laboratory events (Regex + MedGemma + Recovery, 0.841), suggesting that for semantically richer or more variable segments, the LLM-enhanced strategy may better capture contextual cues.

Surgery achieves the highest F1 scores overall: it likely benefit from clearer structural markers and more explicit terminology, which facilitate both segmentation and tagging. In contrast, Biopsy shows the lowest performance across all config-

Event Tag	Regex	Regex + MQ + Recovery	Regex + MG + Recovery	Regex + MQ	Regex + MG
Imaging	**0.815**	0.749	0.812	0.682	0.709
Surgery	**0.923**	0.767	0.788	0.579	0.743
Biopsy	**0.605**	0.465	0.359	0.458	0.340
Treatment	0.697	**0.721**	0.675	0.628	0.659
Laboratory	0.759	0.794	**0.841**	0.817	0.835
Overall	**0.757**	0.668	0.720	0.623	0.652

Table 4: Per-category F1 scores for tag prediction. The Overall row reports micro-averaged F1 across all tags. MQ = MedicalQwen3; MG = MedGemma.

urations, likely because biopsy information is often embedded within surgical descriptions and not clearly separated, making both boundary detection and label assignment more challenging.

5 Conclusions

In this study, we investigated semantics-driven text segmentation as a core component for identifying temporal clinical events in oncological reports. By comparing a fully rule-based approach and a cascaded regex–LLM pipeline with and without a recovery mechanism, we demonstrated how segmentation quality directly impacts downstream event type identification.

The results show that regex-based segmentation ensures superior boundary precision and structural stability, while LLM-enhanced approaches better preserve semantic coherence within segments. These findings support the view that segmentation should be treated as a preprocessing step in order to make reports more easily processable for downstream analysis, but it also consists of a modelling task that enables reliable reconstruction of the patient's clinical trajectory.

As future work, we aim to extend this study by scaling up the evaluation dataset and involving multiple annotators to improve the reliability of the gold standard. We also plan to validate the framework on multiple external, heterogeneous datasets to further assess generalization. Methodologically, future iterations will explore the LLM-only segmentation on the full-report to better isolate the contribution of the recovery mechanism, and conduct an ablation study to decouple the impact of gap-filling and global rollback. Finally, we aim to replace the rule-based event tagger with a transformer-based classifier to evaluate downstream performance.

6 Limitations

While the results are promising, several directions for further improvement emerge. Expanding the evaluation dataset beyond the current 343 segments would enable a more robust assessment of generalization, and involving multiple annotators would strengthen the reliability of the gold standard. From a modeling perspective, evaluating LLMs in full precision (without quantization) could better capture their intrinsic capabilities, while a systematic optimization of the recovery mechanism may further enhance performance. Finally, extending event tagging beyond regex-based rules toward transformer-based approaches represents a natural step toward a more flexible and fully data-driven pipeline.

Ethics Statement

The use of data for this study has been implemented in full compliance with ethics and GDPR requirements. Specifically, data usage has been approved by the Ethics Committee of our hospital to conduct the presented research and the de-identification of sensitive data has been performed. Approval protocol number from the relevant Ethics Committee can be provided on request.

References

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. 2022. Large language models are few-shot clinical information extractors. *Preprint*, arXiv:2205.12689.

Joe Barrow, Raj Patel, Misha Kharkovski, Ben Davies, and Ryan Schmitt. 2025. Safepassage: High-fidelity information extraction with black box llms. *Preprint*, arXiv:2510.00276.

Doug Beeferman, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine learning*, 34(1):177–210.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Joshua Davis, Thomas Sounack, Kate Sciacca, Jessie M Brain, Brigitte N Durieux, Nicole D Agaronnik, and Charlotta Lindvall. 2025. Medslice: Fine-tuned large language models for secure clinical note sectioning. *Preprint*, arXiv:2501.14105.

Joshua Denny, Anderson Spickard, Kevin Johnson, Neeraja Peterson, Josh Peterson, and Randolph Miller. 2009. Evaluation of a method to identify and categorize section headers in clinical documents. *Journal of the American Medical Informatics Association : JAMIA*, 16:806–15.

Hannah Eyre, Alec B Chapman, Kelly S Peterson, Jianlin Shi, Patrick R Alba, Makoto M Jones, Tamara L Box, Scott L DuVall, and Olga V Patterson. 2021. Launching into clinical space with medspacy: a new clinical text processing toolkit in python. *Preprint*, arXiv:2106.07799.

Chris Fournier. 2013a. Evaluating Text Segmentation using Boundary Edit Distance. In *Proceedings of 51st Annual Meeting of the Association for Computational Linguistics*, page to appear, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chris Fournier. 2013b. Evaluating text segmentation using boundary edit distance. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1702–1712.

Markus Frohmann, Igor Sterner, Ivan Vulić, Benjamin Minixhofer, and Markus Schedl. 2024. Segment any text: A universal approach for robust, efficient and adaptable sentence segmentation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11908–11941.

Akshay Goel. 2026. LangExtract.

Aman Jaiswal and Evangelos Milios. 2023. Breaking the token barrier: Chunking and convolution for efficient long text classification with bert. *arXiv preprint arXiv:2310.20558*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Ricardo Kleinlein, Kathryn Gray, David Bates, and Vesela Kovacheva. 2025. Spell-llms: A scalable and privacy-compliant nlp pipeline using locally hosted large language models for clinical information extraction.

Livia Lilli, Carlotta Masciocchi, Antonio Marchetti, Giovanni Arcuri, and Stefano Patarnello. 2025a. Prompting large language models for italian clinical reports: A benchmark study. In *Proceedings of the 24th Workshop on Biomedical Language Processing*, pages 190–200.

Livia Lilli, Stefano Patarnello, Nikola Capocchiano, Carlotta Masciocchi, and Mario Santoro. 2025b. Improving clinical report classification with sentence boundary detection. pages 190–196.

Livia Lilli, Andrea Rosati, Giovanni Paolo Tobia, Massimo Criscione, Federica Tomassini, Chiara Dachena, Alice Luraschi, Chiara Cantarini, Carolina De Maria, Luigi Congedo, Massimo Bernaschi, Stefano Patarnello, and Anna Fagotti. 2026. Prompt-orchestrated large language models for clinical information extraction. *Research Square*. Preprint.

Lev Pevzner and Marti A Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108.

Nipun Sadvilkar and Mark Neumann. 2020. Pysbd: Pragmatic sentence boundary disambiguation. *arXiv preprint arXiv:2010.09657*.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, and 1 others. 2025. Medgemma technical report. *arXiv preprint arXiv:2507.05201*.

William F. Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, and James Pustejovsky. 2014. Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association : JAMIA*, 20.

Risha Surana, Adrian Law, Sunwoo Kim, Rishab Sridhar, Angxiao Han, and Peiyu Hong. 2025. Cnsight: Evaluation of clinical note segmentation tools. *Preprint*, arXiv:2512.22795.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Shin, Kaleb Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony Costa, Mona Flores, Ying

Zhang, Tanja Magoc, Christopher Harle, Gloria Lipori, Duane Mitchell, William Hogan, Elizabeth Shenkman, Jiang Bian, and Yonghui Wu. 2022. A large language model for electronic health records. *npj Digital Medicine*, 5.

Zongxin Yang, Hongyi Yuan, Raheel Sayeed, Amelia Li Min Tan, Enci Cai, Mohammed Moro, Xiudi Li, Huaiyuan Ying, Nicholas Brown, Griffin Weber, Sheng Yu, Isaac Kohane, and Tianxi Cai. 2025b. Clines: Clinical llm-based information extraction and structuring agent. *medRxiv*.