LiveDrill: Multimodal Segment-Triggered Data-to-Text for Time Series Foundation Models

Soumyadipta Sengupta[*†] AIQ

Amine EL KHAIR[*†] AIQ

Sebastiaan Buiting[*†] AIQ

Imane Khaouja [‡] AIQ

Yahia Salaheldin Shaaban[†] AIQ

Abdallah Zakaria Benzine[*†] AIQ

Abstract

Time-series foundation models show strong results on static benchmarks, but their potential in live industrial reporting is only beginning to be explored. In drilling, continuous multivariate sensor streams must be transformed into Daily Drilling Reports (DDRs), where each entry aligns with activity boundaries. Automating this process offers an opportunity to deliver reports that are both timely and consistent, reducing the burden of manual compilation.

We present **LiveDrill**, a streaming pipeline for **multimodal segment-grounded data-to-text generation**. LiveDrill integrates two modules: a **Live Segmentation Module** that detects activity transitions in real time, and a **Multimodal Text Generation Module** that conditions report entries on both sensor signals and the detected segments. This design ensures that generated text is explicitly tied to operational intervals, providing structured updates directly from live data.

Evaluation on large-scale field data demonstrates that LiveDrill can reliably capture stable operations and generate coherent DDR entries. Segment-level metrics reveal the sensitivity of boundary detection, highlighting areas where further improvement can yield even stronger results.

Overall, LiveDrill demonstrates the feasibility of segment-grounded, multimodal reporting in industrial settings. It opens the door for adapting TSFMs beyond static benchmarks toward practical, boundary-sensitive applications where live sensor data must be translated into actionable narratives.

1 Introduction

Industrial operations generate long, multivariate sensor streams that must be condensed into short, actionable reports. In drilling, these take the form of Daily Drilling Reports (DDRs), which record operational activities and are critical for coordination, planning, and safety (example in Appendix 5). Today, DDRs are compiled manually after each 24-hour shift, leading to delays and inconsistencies. Automating this process requires live generation grounded in multimodal signals that define operations. DDR entries are tied to activities such as drilling, tripping, or circulating, each marked by distinct sensor patterns. Without aligning text to these segments, reports lose operational meaning.

We introduce **LiveDrill**, a streaming pipeline that couples time-series segmentation with multimodal text generation. A *Live Segmentation Module* detects activity boundaries directly from sensor streams in real time, while a *Multimodal Text Generation Module* produces DDR entries conditioned jointly on sensor dynamics and the detected segment. This design ensures that generated text remains timely and explicitly tied to drilling activities.

[*] **Equal Contributions**

[†] firstname.lastname@aiqintelligence.ae

[‡] KhaoujaI@gmail.com

NeurIPS 2025 Workshop on Recent Advances in Time Series Foundation Models (BERT^{2}S).

Time-series foundation models (TSFMs) show strong performance on static benchmarks [Ansari et al., 2024, Goswami et al., 2024, Woo et al., 2024], but live industrial reporting introduces unique demands: detecting activity boundaries, aligning multimodal signals with language, and generating text at low latency. Prior work in drilling automation has explored anomaly detection [Benzine et al., 2024b], DDR codification [Benzine et al., 2024c], and sensor–text alignment for retrieval and zero-shot description [Buiting et al., 2025]. TSFMs have also demonstrated cross-domain generalization [Khaouja et al., 2025, Benzine et al., 2024a]. LiveDrill builds on these directions, providing a multimodal benchmark for segment-grounded live data-to-text generation, highlighting both opportunities and limitations of TSFMs in noisy, boundary-sensitive industrial settings.

2 Method

LiveDrill is a streaming pipeline with two modules: a *Live Segmentation Module* that detects activity changes and a *Multimodal Text Generation Module* that generates text for the detected region. Figure 1 shows the workflow.

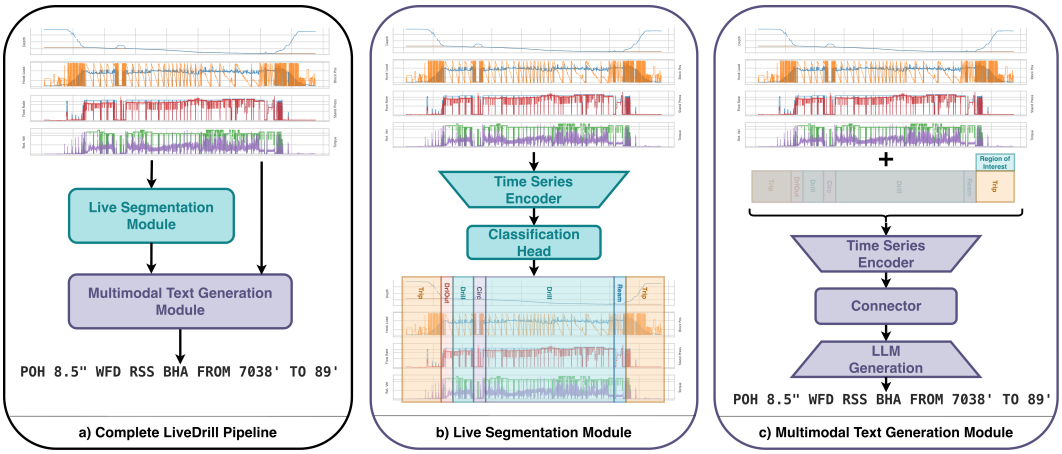


Figure 1: **LiveDrill pipeline.** (a) End-to-end framework for real-time drilling activity segmentation and description. (b) *Live Segmentation Module:* A time-series encoder and classification head detect activity segments. (c) *Multimodal Text Generation Module:* Encoded time-series are passed through a connector and LLM to generates a segment-grounded DDR entry only for the region of interest, corresponding to the last complete segment generated by the LSM.

2.1 Live Segmentation Module (LSM)

The LSM, illustrated in Figure 1.a, is a live time-series segmentation model that performs multi-class point-wise classification on the incoming sensor stream. Its goal is to assign an activity label to every time step and to detect transitions between activities as they happen. The LSM is implemented as a time-series foundation model (Moment and Moirai Large) adapted for segmentation [Khaouja et al., 2025].

2.2 Multimodal Text Generation Module (MTGM)

Instead of generating text continuously, the MTGM is triggered only after the LSM raises a change event, meaning when a segment is complete. The MTGM, illustrated in Figure 1.b, then generates a DDR entry for this finished segment. The MTGM has four components.

Region-of-Interest (ROI) Selector. The ROI selector takes the segmentation masks produced by the LSM and identifies the activity segment that requires a new DDR entry. It generates a binary mask aligned with the sensor stream: time steps inside the detected segment are marked with 1, and all others with 0. This mask serves two purposes: it explicitly delimits the segment of interest, and it is passed to the next component (Time-Series Encoder) as an additional input channel. By grounding subsequent encoding in the segment identified by the LSM, the ROI selector ensures that text generation is tied to meaningful intervals.

Time-Series Encoder (TSE). The TSE converts the input into a sequence of embeddings optimized for text generation. Its input is the raw sensor channels concatenated with the binary ROI mask.

Formally, let $\mathbf{X}_{\text{TSE}} \in \mathbb{R}^{T\times(C+1)}$ denote the encoder input, where C is the number of sensor channels and the extra channel corresponds to the binary ROI mask. This design provides the encoder both with the sensor dynamics and an explicit signal about which region to prioritize. Moment and Moirai Large models were chosen here from experience in previous study [Khaouja et al., 2025, Buiting et al., 2025].

TSE-to-LLM Connector. A linear projection maps the TSE embedding dimension to $\mathbf{H}$, in the LLM token-embedding space.

LLM Decoder. The LLM receives $\mathbf{H}$ and generates the DDR entry autoregressively. We chose Phi3-mini-4K-instruct here.

2.3 Model Training Procedures

LSM Training. We initialize the LSM from a pretrained TSFM and fine-tune on annotated sequences $\{(\mathbf{X}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{N}$. The loss is point-level cross-entropy on activity labels.

MTGM Training. During training, the LLM is frozen. The TSE and the connector are trainable. Given a detected segment, the TSE outputs $\mathbf{Z}_{\text{ROI}}$, which the connector projects into the LLM embedding space. Conditioned by the projected features, the LLM predicts the DDR text:

$$\mathcal{L}_{\text{NT}} = -\sum_{t=1}^{M} \log p_\theta(w_t \mid w_{<t}, \mathbf{Z}_{\text{ROI}}),$$

where w_t is the t-th token and M is the entry length.

3 Evaluation

3.1 Dataset

We evaluate on a large-scale industrial dataset comprising daily drilling data from over 100 active drilling rigs, collected over a period of 3 years. This dataset, which includes 1810 distinct drilling phases (split into 1353 for training, 101 for validation, and 356 for testing), contains a total of over 35,987 days of continuous multivariate streams from surface sensors, sampled at 1 Hz. See Appendix A for more details.

For training and validation of the Live Segmentation Module (LSM), a subset of this data has been annotated with activity labels derived from manually written Daily Drilling Reports (DDRs), resulting in over 68,017 distinct activity segments. Each segment in these DDRs contains an activity description, which serves as the reference text for the Multimodal Text Generation Module (MTGM)

3.2 Evaluation Tasks

We assess streaming segmentation quality across drilling activities, the quality of DDR generation for segment-grounded entries, and the overall system performance in live settings.

3.3 Metrics

For segmentation, we report both segment-based $F1_{IoU}$ (Khaouja et al. [2025]) and point-wise $F1_{pw}$. The segment-based $F1_{IoU}$ evaluates detection quality by matching predicted and ground-truth segments based on their intersection-over-union, while $F1_{pw}$ measures classification accuracy at the point level, independent of segment boundaries. For DDR generation, an automated LLM judge (Benzine et al. [2025]) (Llama-3.3-70B-instruct 4 bit) scores operational accuracy, depth consistency, and language quality. The overall system score is computed as the harmonic mean of the segmentation and generation scores.

3.4 LSM performance

Segmentation Model	Overall		CM		CMT		CORE		CSG		DRILL		DRLOUT		REAM		STKP		TRIP	
	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$	$F1_{IoU}$	$F1_{pw}$
Moment-large	0.467	0.827	0.094	0.823	0.395	0.740	0.533	0.935	0.365	0.844	0.526	0.942	0.423	0.736	0.446	0.743	0.000	0.923	0.567	0.818
Moirai-large	0.484	0.809	0.233	0.765	0.394	0.739	0.182	0.986	0.298	0.746	0.538	0.950	0.415	0.710	0.489	0.739	0.000	0.907	0.565	0.790

Table 1: $F1_{IoU}$ and $F1_{pw}$ (overall + per class) for segmentation results of Moment and Moirai models.

The results in Table 1 show a clear gap between point-wise accuracy ($F1_{pw}$ around 0.81) and segment-level performance ($F1_{IoU}$ around 0.47). The segment metric is highly sensitive to small interruptions: even short misalignments split otherwise correct segments and lower scores. Manual inspection also revealed short interrupting segments in the ground truth, suggesting that part of the gap comes from annotation noise rather than model errors.

Overall, point-wise accuracy confirms that most time steps are correctly labeled, especially for stable activities like **DRILL**, **DRLOUT** and **TRIP**. More variable or rare classes such as **STKP** remain harder to capture (see Appendix A.3 for all class definitions). Between the two models, Moirai-large provides slightly better segment-level alignment, while both show similar point-wise accuracy. These results indicate that LiveDrill's LSM achieves reliable classification, but progress will require smoothing predictions and improving rare-class detection to close the gap between point-wise and segment-level scores.

3.5 MTGM performance

TSE Model	Avg LLM [0-1]	CM	CMT	CORE	CSG	DRILL	DRLOUT	REAM	STKP	TRIP
Moment-large	0.359	0.137	0.333	0.291	0.353	0.590	0.456	0.499	0.114	0.200
Moirai-large	0.340	0.212	0.344	0.309	0.354	0.563	0.368	0.454	0.210	0.157

Table 2: LLM judge evaluation scores (normalized to [0–1]) for Moment and Moirai models.

LSM	TSE: Moment-large			TSE: Moirai-large		
	Seg $F1_{IoU}$	Avg LLM	Combined	Seg $F1_{IoU}$	Avg LLM	Combined
Moment-large	0.467	0.359	0.405	0.467	0.340	0.394
Moirai-large	0.484	0.359	0.412	0.484	0.340	0.399

Table 3: Combined evaluation of segmentation and text generation. The "Combined" column is the harmonic mean.

Table 2 shows moderate overall performance (0.34–0.36), with clear variability across activities. Frequent and stable operations such as **DRILL**, **DRLOUT**, and **REAM** reach higher scores ($>$0.45), while rare or complex ones like **STKP**, **CM**, and **TRIP** remain difficult.

Moment-large performs better in continuous drilling phases, whereas Moirai-large shows relative strength in short or transition-heavy activities. These results highlight the need for better handling of rare events and more robust domain grounding in text generation.

3.6 Full system performance

The results in Table 3 indicate that overall system performance remains moderate when combining segmentation and generation. The harmonic mean shows how deficiencies in either module significantly reduce the joint score: the best configuration reaches 0.412 when using Moirai for segmentation and Moment for generation, slightly outperforming Moment-only (0.405) and Moirai-only (0.399) setups. This suggests that segmentation quality from Moirai provides a small but consistent advantage, while Moment remains stronger in time-series encoding for text generation.

LiveDrill produces coherent updates in stable phases such as **DRILL**, but struggles in transition-heavy or rare operations. Qualitative results of LiveDrill are shown in Appendix B and C. For inference timings see Appendix E.4.

4 Conclusion

This work introduced **LiveDrill**, a streaming system for segment-triggered data-to-text generation in industrial time-series. By combining a Live Segmentation Module with a Multimodal Text Generation Module, the system shifts DDR creation from delayed, end-of-day summaries to continuous, real-time reporting.

Our evaluation on large-scale drilling data shows that LiveDrill can reliably classify stable operations and generate coherent entries, providing engineers with timely and structured information. At the same time, the results highlight current limitations: segmentation errors propagate into generation, transition-heavy activities remain difficult to model, and absolute generation scores leave room for improvement.

These findings demonstrate both the feasibility and challenges of live industrial reporting. Future work should focus on smoothing segmentation boundaries, improving handling of rare and complex operations, and enhancing domain grounding in text generation. Beyond drilling, the approach of segment-grounded live reporting may extend to other industrial domains where continuous sensor data must be converted into actionable narratives.

References

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Abdallah Benzine, J.S. Buiting, Soumyadipta Sengupta, Badal Gupta, and Youssef Tamaazousti. When larger isn't better: Lightweight CNNs outperform large time-series models in classification of oil and gas drilling data. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024a. URL `https://openreview.net/forum?id=at9c42t6A2`.

Abdallah Benzine, Amine El Khair, Sebastiaan Buiting, Soumyadipta Sengupta, Badal Gupta, Ufaq Khan, Youssef Tamaazousti, Sudheesh Vadakkekalam, Sreejith Balakrishnan, Ahmed Jhinaoui, Imane Chraibi, Dhaker Ezzeddine, Arghad Arnaout, Shreepad Purushottam Khambete, Paulinus Bimastianto, Abdullah Ibrahim, and Ahmed Al Hai. Ai-automated drilling rtoc with ml and deep learning anomaly detection approach. In *Abu Dhabi International Petroleum Exhibition and Conference (ADIPEC)*, Abu Dhabi, UAE, 2024b. Society of Petroleum Engineers. doi: 10.2118/222517-MS.

Abdallah Benzine, Youssef Tamaazousti, Sudheesh Vadakkekalam, Sreejith Balakrishnan, Imane Chraibi, Dhaker Ezzeddine, Arghad Arnaout, Shreepad Purushottam Khambete, Paulinus Bimastianto, Shahid Duivala Muhammad, Idrees Muhammad Mughal, and Abdulrahman Abduljalil Murad. Ai-automated codification-qc model for daily drilling reports. In *Abu Dhabi International Petroleum Exhibition and Conference (ADIPEC)*, Abu Dhabi, UAE, 2024c. Society of Petroleum Engineers. doi: 10.2118/222675-MS.

Abdallah Benzine, Soumyadipta Sengupta, Sebastiaan Buiting, Imane Khaouja, Yahia Salaheldin Shaaban, and Amine EL KHAIR. LLMs as judges for domain-specific text: Evidence from drilling reports. In *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*, 2025. URL `https://openreview.net/forum?id=zK2akNuseU`.

Sebastiaan Buiting, Soumyadipta Sengupta, Abdallah Benzine, Amine El Khair, Imane Khaouja, and Youssef Tamaazousti. Drimm: Drilling multimodal model for time-series and text. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, Vancouver, Canada, 2025. PMLR.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv preprint arXiv:2402.03885*, 2024.

Imane Khaouja, Amine EL KHAIR, Abdallah Benzine, Sebastiaan Buiting, Soumyadipta Sengupta, and Youssef Tamaazousti. Do large foundation models improve time series segmentation? an industrial case study in oil and gas drilling. In *1st ICML Workshop on Foundation Models for Structured Data*, 2025. URL `https://openreview.net/forum?id=LcAZkhb9uz`.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. 2024.

A Appendix - Data

A.1 Features

The eight key features used in the input data capture drilling information:

- **Bit Depth:** The measured depth of the drill bit within the hole.
- **Hole Depth:** The total depth of the drilled hole.
- **Hook Load:** The weight supported by the hook that holds the drilling assembly that go in the hole.
- **Block Position:** The vertical position of the above-ground block that holds the hook and drilling assembly that go in the hole.
- **Standpipe Pressure:** The fluid pressure inside the drilling pipe.
- **Rotary Speed:** The rotational speed of the drilling assembly in the hole.
- **Flow Rate:** The rate at which drilling fluid is pumped into the well.
- **Torque:** The rotational force applied to the drilling assembly during drilling operations.

A.2 Time Series Pre-Processing

- Sensor signals are resampled to 1 Hz.
- Gaps shorter than five seconds are linearly interpolated; longer gaps are left as nulls.
- Noise is reduced using domain thresholds followed by Z-score filtering on sliding windows.
- Features are normalized with expert-defined physical limits for cross-well consistency.
- To match TSFM pretraining input size, time-series are finally sub-sampled to 512 points.

A.3 DDR Activity Class Definitions

Table 4 provides concise definitions for the main Daily Drilling Report (DDR) activity classes used to train and evaluate the Live Segmentation Module (LSM).

Table 4: Definitions of DDR Activity Class Codes

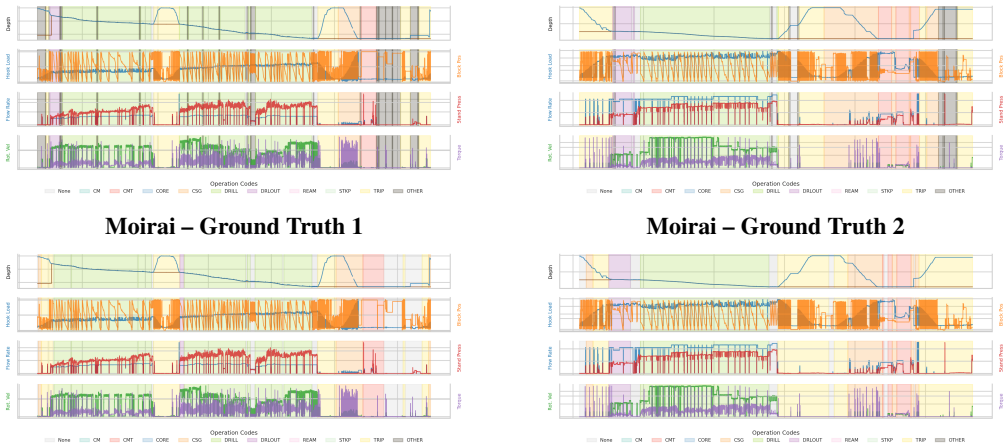
Code	Full Name	Description
CM	Completion	Activities related to preparing the well for production, including installing completion equipment and performing tests to ensure flow from the reservoir.
CMT	Cementing	Pumping cement into the wellbore to secure steel casing (pipe) or to seal off specific geological zones.
CORE	Coring	Cutting and retrieving a cylindrical rock sample (a "core") from the bottom of the well for geological analysis.
CSG	Running Casing	Lowering and assembling sections of large-diameter steel pipe (casing) into the newly drilled hole to provide structural integrity.
DRILL	Drilling	The primary activity of creating a new hole by rotating the drill bit and applying weight to break the rock formation.
DRLOUT	Drilling Out	Drilling through cement plugs or other obstructions that were intentionally left inside the existing casing.
REAM	Reaming	Enlarging or smoothing a previously drilled section of the wellbore to ensure it is the correct diameter and to reduce friction.
STKP	Stuck Pipe	An unplanned event where the drill string becomes immobilized in the well, preventing movement or rotation.
TRIP	Tripping	The operation of pulling the entire drill string out of the hole (trip out) or running it back in (trip in), typically to change the drill bit.

B Qualitative Results - LSM

This section provides qualitative examples of predicted activity segmentation compared to ground truth annotations for two models: **Moirai-Large** (the best-performing LSM) and **Moment-Large**. These examples illustrate how the models handle activity transitions and maintain segment-level alignment in real drilling scenarios.

B.1 Moirai-Large

Figure 2 presents qualitative results for Moirai-Large. The top rows in each subfigure correspond to ground truth annotations, which include the class `OTHER` to mask surface activities. The bottom rows show predictions made by the LSM. Moirai-Large effectively identifies activity boundaries and demonstrates strong alignment with the ground truth across complex transitions.



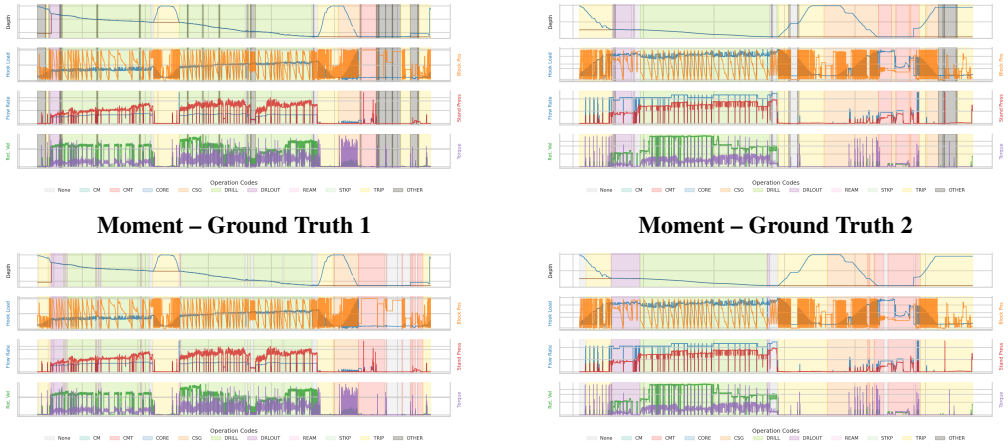
Moirai – Ground Truth 1 **Moirai – Ground Truth 2**

Moirai – LSM Predictions 1 **Moirai – LSM Predictions 2**

Figure 2: Qualitative examples comparing ground truth and Moirai-Large predictions. Ground truth annotations (top) include `OTHER` to mask surface activities, while predictions (bottom) illustrate the LSM's ability to capture activity transitions and segment boundaries.

B.2 Moment-Large

Similarly, Figure 3 shows qualitative results for Moment-Large. As with Moirai-Large, ground truth annotations appear on the top rows, and model predictions are on the bottom. Although Moment-Large captures major transitions, it tends to produce less precise segment boundaries compared to Moirai-Large.



Moment – Ground Truth 1 **Moment – Ground Truth 2**

Moment – LSM Predictions 1 **Moment – LSM Predictions 2**

Figure 3: Qualitative examples comparing ground truth and Moment-Large predictions.

C Qualitative Results - MTGM

Best model, Moment-Large, is used as MTGM in the below results.

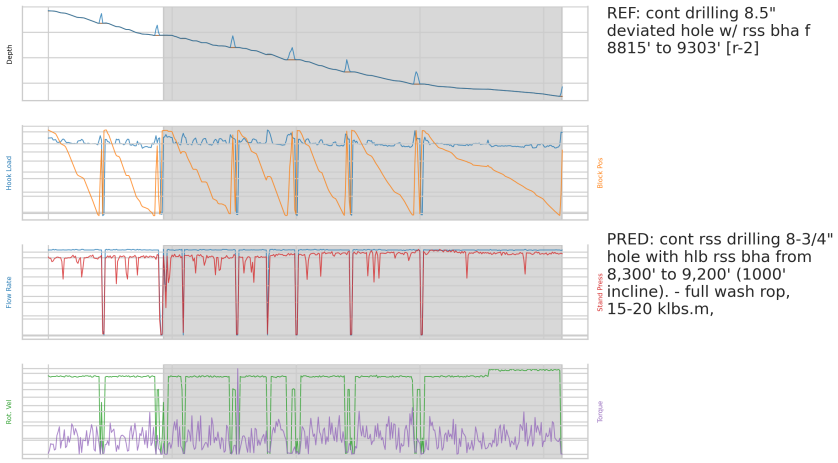


Figure 4: MTGM-generated DDR entry for a **DRILL** activity segment.

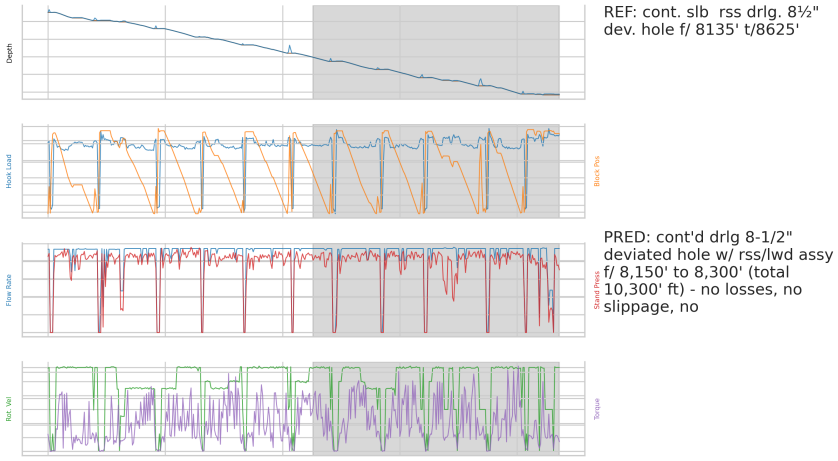


Figure 5: Another MTGM-generated DDR entry for a **DRILL** activity segment.

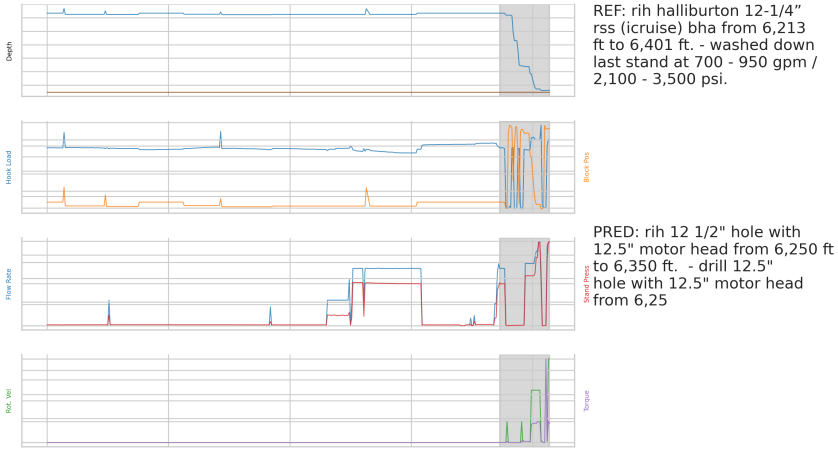


Figure 6: MTGM output for a **TRIP** activity segment.

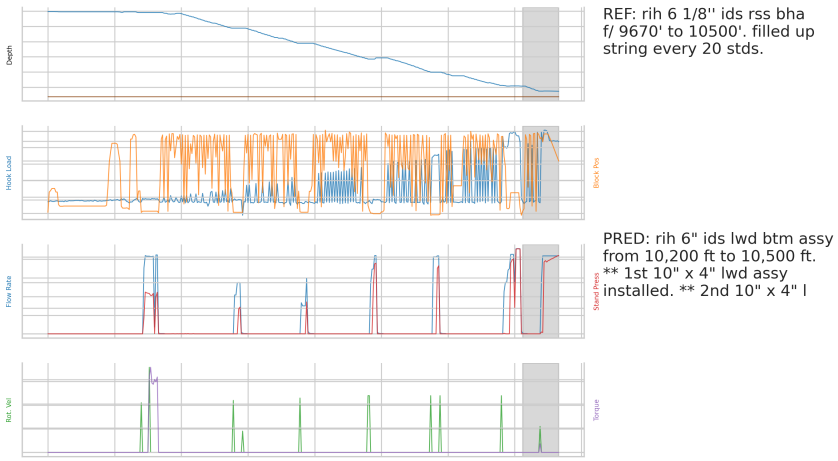


Figure 7: Another example of MTGM-generated DDR for a **TRIP** activity segment.

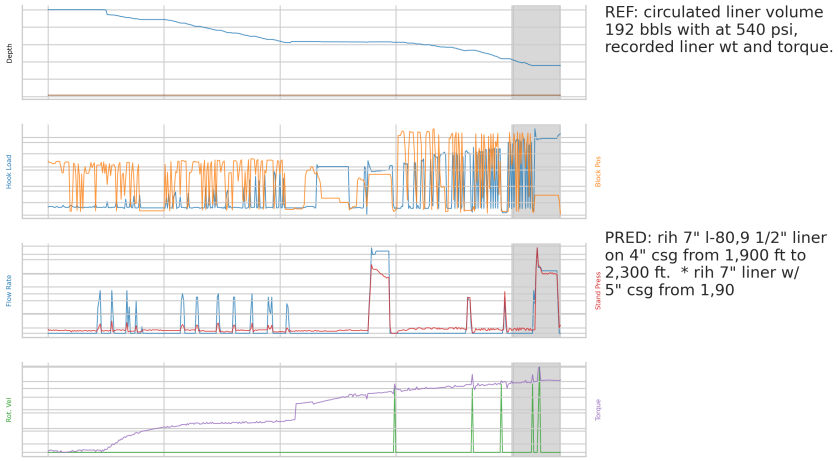


Figure 8: MTGM output for a **CSG** activity segment.

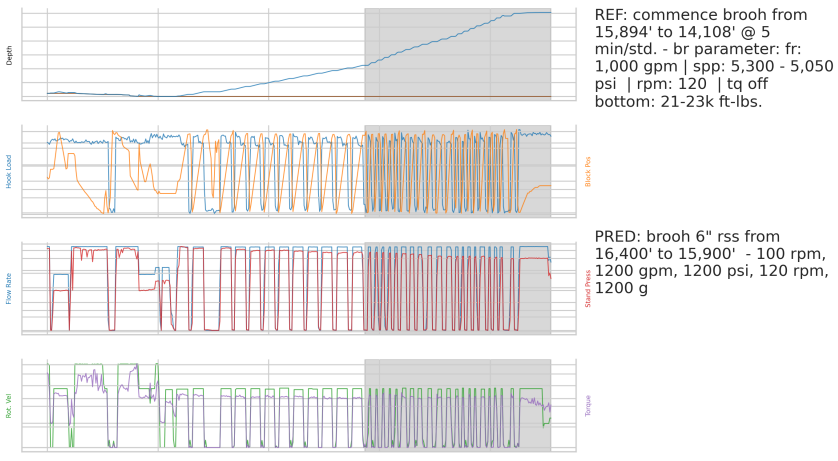


Figure 9: MTGM-generated DDR entry for a **REAM** activity segment.

D Appendix - Example DDR

Table 5 provides a detailed example of a Daily Drilling Report (DDR), illustrating time-stamped activities, anomalies, and sensor values.

Table 5: Example Daily Drilling Report (DDR)

Start	End	Code	Operation
00:00	01:45	TRIP	RIH with BHA to 2140 m, tagged bottom, spaced out, pumped to stabilize
01:45	02:30	CIRC	Circulated bottoms up, SPP 2400 psi, clean returns
02:30	05:30	DRILL	Drilled 12-1/4" hole 2140–2185 m, WOB 20 klbs, torque up to 12 kft-lb
05:30	06:30	REAM	Reamed 2140–2185 m, reduced drag, torque normalized
06:30	07:15	CIRC	Conditioned hole, minor gas at shaker, flow rate +10%
07:15	09:45	DRILL	Drilled to 2220 m, partial losses 50 gpm observed
09:45	12:15	STKP	Pipe stuck at 2215 m, no rotation; worked string, spotted LCM, freed
12:15	13:30	CIRC	Conditioned hole for casing, stable returns, flow check good
13:30	16:30	CSG	Ran 9-5/8" casing to 2220 m, tight at 2210 m, worked through
16:30	24:00	CMT	Cemented casing; 40 bbl returns lost; WOC for remainder of day

E Appendix - LiveDrill pipeline modules

This section details the internal structure and operation of the two core components of LiveDrill: the **Live Segmentation Module (LSM)**, which determines *when to write*, and the **Multimodal Text Generation Module (MTGM)**, which determines *what to write*. Together, these modules enable real-time, segment-grounded text generation from continuous multivariate sensor streams.

E.1 Live Segmentation Module (LSM)

The LSM continuously processes the incoming time-series data to assign operational activity labels (e.g., `DRILL`, `CIRC`, `TRIP`) at each timestep and to detect transitions between them. Each change in predicted label defines a segment boundary that triggers text generation.

Architecture. The LSM is implemented as a multichannel time-series encoder followed by a dense classification head. We use pretrained time-series foundation models (Moment-Large and Moirai-Large) adapted for segmentation. Each encoder processes the input stream $\mathbf{X} \in \mathbb{R}^{T \times C}$ and outputs per-step activity logits:

$$\hat{\mathbf{y}}_t = \text{softmax}(f_\theta(\mathbf{X}_{1:t})).$$

Boundaries are confirmed when $\hat{y}_t \neq \hat{y}_{t-1}$, with a short temporal smoothing window to prevent spurious triggers.

Output. The LSM produces:

- Continuous activity label stream $\{\hat{y}_t\}$,
- Boundary timestamps $\{\tau_i\}$, and
- Corresponding data segments $S_i = \mathbf{X}_{\tau_{i-1}+1:\tau_i}$ passed to the MTGM.

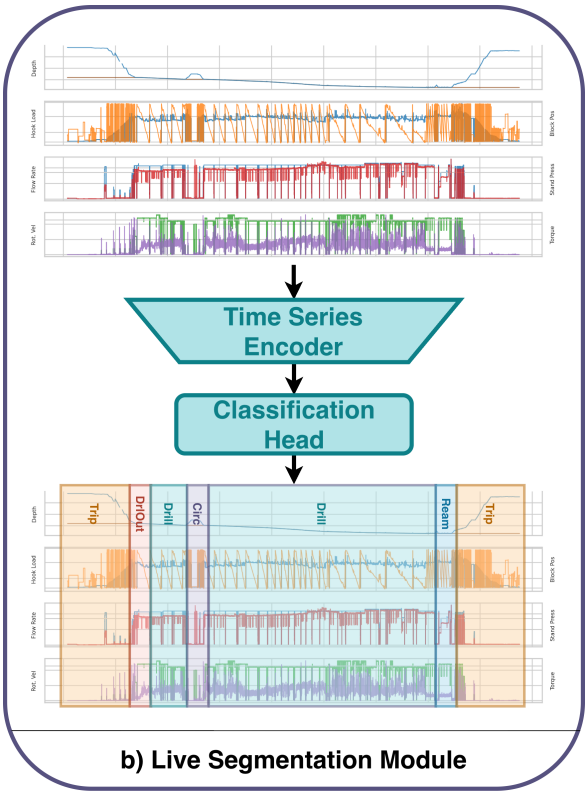


Figure 10: **Live Segmentation Module.** A time-series encoder and classification head monitor the sensor stream in real time, assigning per-step activity labels and triggering report generation upon segment completion.

E.2 Multimodal Text Generation Module (MTGM)

The MTGM is triggered each time the LSM detects a new activity. It generates a concise DDR entry summarizing the completed segment, grounding language generation in sensor data context.

Region-of-Interest (ROI) Selector. The ROI Selector receives segmentation masks from the LSM and constructs a binary mask aligned with the sensor stream:

$$m_t = \begin{cases} 1, & t \in [\tau_{i-1} + 1, \tau_i] \\ 0, & \text{otherwise.} \end{cases}$$

This mask explicitly defines the segment of interest and is concatenated with the sensor channels for time-series encoding.

Time-Series Encoder (TSE). The encoder receives $\mathbf{X}_{\text{TSE}} = [\mathbf{X} \,\|\, m] \in \mathbb{R}^{T \times (C+1)}$ and converts it into temporal embeddings $\mathbf{Z} \in \mathbb{R}^{T \times d}$. Moment-Large and Moirai-Large were used as pretrained TSEs, providing general-purpose temporal features transferable across drilling operations.

TSE-to-LLM Connector. A linear projection maps $\mathbf{Z}$ to the LLM token-embedding space $\mathbf{H}$:

$$\tilde{\mathbf{Z}} = W\mathbf{Z} + b, \quad \tilde{\mathbf{Z}} \in \mathbb{R}^{T \times H}.$$

Only the connector and TSE are trainable; the LLM remains frozen.

LLM Decoder. The Large Language Model (Phi-3-mini-4K-instruct in this work) generates text conditioned on $\tilde{\mathbf{Z}}$:

$$\hat{y}_i = \arg\max_{w_{1:M}} \prod_{t=1}^{M} p_\theta(w_t \mid w_{<t}, \tilde{\mathbf{Z}}),$$

producing a segment-grounded DDR entry such as: *"Drilled 12¼" hole from 2140 m to 2185 m, WOB 20 klbs, torque 12 kft-lb."*

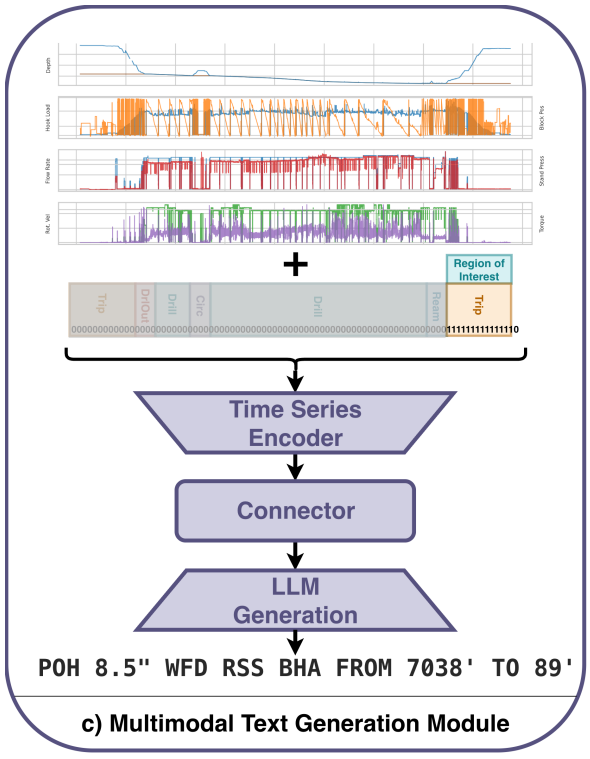


Figure 11: **Multimodal Text Generation Module.** Encoded sensor data and the ROI mask are projected to the LLM embedding space, conditioning text generation on the detected operational segment.

E.3 Streaming Execution Loop

At inference time, the full LiveDrill pipeline operates continuously:

1. The LSM monitors sensor inputs and detects an activity transition.
2. Upon boundary confirmation, the completed segment S_i and ROI mask are sent to the MTGM.
3. The MTGM encodes the segment, projects features to the LLM space, and generates the textual entry.
4. The generated entry is appended to the live DDR log.

This modular, segment-triggered design decouples temporal segmentation (*when to write*) from language generation (*what to write*), ensuring interpretability, low latency, and scalability for real-time industrial reporting.

E.4 Inference and Latency

To validate the "live" claim of the pipeline, we benchmarked the inference latency of its core components. These benchmarks were run on the best-performing configuration identified in Section 3 (Moirai-large for the LSM and Moment-large for the MTGM's TSE) on a single V100 GPU. The pipeline's design distinguishes between continuous real-time processing (LSM) and event-triggered generation (MTGM). The results are summarized in Table 6.

Live Segmentation Module (LSM). The LSM operates continuously, processing the incoming 1 Hz sensor stream. Its average inference time for a single segmentation step is **192 ms**. This is significantly faster than the 1000 ms (1 Hz) data interval, allowing the LSM to run in real-time without accumulating a processing backlog.

Multimodal Text Generation Module (MTGM). The MTGM is **not** run continuously. It is triggered only once a complete activity segment is detected by the LSM. In our dataset, the average ground-truth segment duration is **51.5 minutes**, meaning the MTGM will ideally be triggered each 51.5 minutes on average. Upon this trigger, the MTGM execution involves two steps:

- The **Time-Series Encoder (TSE)** processes the full segment's data, taking **325 ms** on average.
- The **LLM Generator** autoregressively produces the text, taking on average **415 ms** for a new DDR entry. This varies depending on the length of the generated DDR entry.

Full LiveDrill. It is crucial to distinguish between the pipeline's two operating states. For the **overwhelming majority** of operations (i.e., at every 1 Hz timestep), the system is processing the continuous sensor stream using only the LSM, resulting in a low latency of approximately **192 ms**. The full pipeline, which generates a new DDR entry, is triggered only upon the completion of an entire activity segment. As noted, this is an infrequent event, occurring on average only **once every 51.5 minutes**. In this specific case, the total latency—from the final sensor input that completes the segment to the generation of the text report—is **0.93 seconds** (931.77 ms). This sub-second timing for final report generation, combined with the low-latency continuous processing, comfortably meets the requirements for live industrial reporting.

Table 6: Inference timings of LiveDrill Modules

Module	Component	Trigger	Avg. Latency (ms)
LSM	TSFM Encoder + Classifier	Continuous (per timestep)	**191.77**
MTGM	Time-Series Encoder (TSE)	Event (per segment)	325.49
	LLM Generator (Phi-3)		414.49
	Total MTGM Latency		**739.98**
Total Latency (with event completion trigger)			**931.77**