Leveraging LLMs for Multimodal Medical Time Series Analysis

Nimeesha Chan* NCHAN19@JHU.EDU
Department of Civil and Systems Engineering
Johns Hopkins University
Baltimore, MD, USA

Felix Parker* FPARKER9@JHU.EDU
Department of Civil and Systems Engineering
Johns Hopkins University
Baltimore, MD, USA

William Bennett WBENNE10@JH.EDU
Department of Anesthesiology and Critical Care Medicine
Johns Hopkins University
Baltimore, MD, USA

Tianyi Wu TWU61@JHU.EDU
Department of Applied Mathematics and Statistics
Johns Hopkins University
Baltimore, MD, USA

Mung Yao Jia MJIA8@JHU.EDU
Department of Computer Science
Johns Hopkins University
Baltimore, MD, USA

James Fackler JIM@JHMI.EDU
Department of Anesthesiology and Critical Care Medicine
Johns Hopkins University
Baltimore, MD, USA

Kimia Ghobadi KIMIA@JHU.EDU
Department of Civil and Systems Engineering
Johns Hopkins University
Baltimore, MD, USA

Abstract

The complexity and heterogeneity of data in many real-world applications pose significant challenges for traditional machine learning and signal processing techniques. For instance, in medicine, effective analysis of diverse physiological signals is crucial for patient monitoring and clinical decision-making and yet highly challenging. We introduce MedTsLLM, a general multimodal large language model (LLM) framework that effectively integrates time series data and rich contextual information in the form of text to analyze physiological signals, performing three tasks with clinical relevance: semantic segmentation, boundary

[0] *These authors contributed equally to this work.

detection, and anomaly detection in time series. These critical tasks enable deeper analysis of physiological signals and can provide actionable insights for clinicians. We utilize a reprogramming layer to align embeddings of time series patches with a pretrained LLM's embedding space and make effective use of raw time series, in conjunction with textual context. Given the multivariate nature of medical datasets, we develop methods to handle multiple covariates. We additionally tailor the text prompt to include patient-specific information. Our model outperforms state-of-the-art baselines, including deep learning models, other LLMs, and clinical methods across multiple medical domains, specifically electrocardiograms and respiratory waveforms. MedTsLLM presents a promising step towards harnessing the power of LLMs for medical time series analysis that can elevate data-driven tools for clinicians and improve patient outcomes.[1]

1. Introduction

Precision medicine and personalized decision support tools have long aimed to leverage multimodal patient data, from free-form text notes to semi-structured electronic health records (EHR) to high-frequency physiological signals, to better capture the complex, high-dimensional patient state and the provider responses. However, combining these heterogeneous data types has been challenging, with early approaches building bespoke models for single data types and tasks. While the advent of transformer architectures enabled deeper insight from merging modalities, it also required meticulous feature engineering and alignment.

We propose utilizing the knowledge and higher-level reasoning that large language models (LLMs) acquire during pretraining to interpret multidimensional, high-frequency physiological signals and produce high-fidelity output. LLMs, trained on vast datasets and adaptable to various downstream tasks, have ushered in a new era of multimodal foundation models (Yin et al., 2023). While quick to be adopted in other domains, healthcare has lagged behind, partly due to a lack of high-quality labeled datasets. Current medical LLMs mostly focus on image-text pairs (Ghosh et al., 2024), EHRs (Li et al., 2024), or clinical notes (Jung et al., 2024). Only recently have LLMs been explored with physiologic signals, and usually for classification (Liu et al., 2024) or report generation (Wan et al., 2024).

The clinical motivation for this work is to leverage the power of large language models to find hidden patterns in time series data. Certainly, single-dimensional time series, such as temperature graphs and pulse oximeter waveforms are comfortably analyzed. However, these "simple" single-dimensional time series can only be analyzed by clinicians as instantaneous "snapshots" and longitudinal patterns are lost (e.g., rates of change or even simply, counts over time of like events). It becomes increasingly challenging for clinicians to extract nuanced, meaningful patterns in time series data when the data is overwhelmingly large and patterns are necessarily multi-dimensional, for instance, cardio-pulmonary interactions that can only be detected with simultaneous analysis of blood pressure waveforms, ventilator waveforms, electrocardiogram (ECG), and even the lower fidelity time series dimensions available in laboratory data. Harnessing the depth of pattern recognition offered by large language models can pave the way to a deeper understanding of clinically significant patterns across multi-dimensional physiological data.

[1] Code for our system is available here: https://github.com/flixpar/med-ts-llm

By aligning multivariate time series with patient context using a patch reprogramming layer, our unified framework performs clinically useful tasks like semantic segmentation, boundary detection, and anomaly detection. At a high level, boundary detection splits signals into periods like breaths or beats. Semantic segmentation further splits time series into distinct, meaningful segments. Anomaly detection identifies periods within the signals that deviate from normal. These tasks are critical for interpreting waveforms, such as those from ECGs, breathing, and other vital signals, and inherently require domain-specific knowledge for phase identification, rendering a one-size-fits-all method impractical, and highlighting the need for integrating unstructured data. Compounded to that, medical signals often contain interactions between different systems (e.g. cardio-pulmonary interactions) that need to be considered in conjunction with each other. To better handle these covariates, we redesign the structure of the data at various points in our architecture, propose multiple covariate-handling methods, and investigate their tradeoffs in performance and accuracy.

We primarily focus on mechanical ventilators to segment breath phases to provide insights into respiratory mechanics, which can guide decisions on weaning or escalating care for these patients. We further test and validate our models on a set of publicly available medical datasets to ensure our framework generalizes across well-studied ECG domains, as well as lesser-studied respiratory waveforms. Our model outperforms state-of-the-art methods in segmenting ECGs and ventilator waveforms, and in detecting arrhythmias. Such capabilities enable downstream applications that could transform critical care medicine, e.g., identifying complex ECG patterns for early diagnosis of life-threatening conditions, or accurate ECG segmentation to measure prognostic parameters such as heart rate variability and QT interval. The multimodal nature of MedTsLLM, along with explicit considerations of covariates, provides an opportunity for further use of medical data to gain more comprehensive insights. In this paper, our main contributions include:

- We propose a framework that uses the power of a pretrained LLM to harness learnings from unstructured and semi-structured data through natural language, with high-dimensional time series signals.
- We design novel methods to more holistically capture the strong correlations between covariates in time series, and discuss their applicability in various settings.
- We construct prompts that provide the LLM with patient-specific information, alongside dataset description, task instructions and sample statistics–all of which contextualizes the provided time series.
- We introduce three novel time series tasks in the space of using text-time series fusion LLMs: boundary detection, semantic segmentation, and anomaly detection.
- Our model outperforms state-of-the-art baselines, including transformer-based models, LLM-based models, traditional time series analysis models, and domain-specific methods, across multiple medical and non-medical tasks and datasets.

Generalizable Insights about Machine Learning in the Context of Healthcare

The integration of advanced computational methods, such as large language models (LLMs), with medical time series analysis holds immense promise for advancing data-driven support tools for clinical decision-making and ultimately improving patient outcomes. The main

advantages of employing LLMs for medical tasks are to leverage the extensive knowledge and reasoning abilities of LLMs and to capitalize on the potential of multimodal data in healthcare. Our model achieves superior performance on both typical time series analysis tasks like anomaly detection and more specialized, clinically insightful tasks like segmentation across multiple medical applications and datasets. This generalizability of our framework to several tasks and medical applications is promising evidence that LLMs may be useful for diverse tasks in healthcare.

Our ablation studies demonstrate that our LLM-based framework effectively utilizes both time series data and text information. Notably, we show that including patient-specific information in the text prompt improves model performance. Thus, LLMs enable the use of natural language prompts as a medium to combine both unstructured and semi-structured information, as is common in EHRs. We develop a range of techniques to better capture the relationships between multivariate physiological signals, and our findings indicate that how covariates are handled significantly impacts model performance. In conclusion, our work leverages the power of LLMs and their ability to integrate diverse data types, which will open up new avenues for more comprehensively consolidating patient information to perform clinically useful downstream tasks, ultimately leading to more accurate diagnoses, targeted treatments, and improved patient care.

2. Related Work

There is a rich and fast-growing literature on the use of large language models and on the use of data in healthcare tasks. We focus on the two most relevant fields of using LLMs for time series analysis, both general and medical time series, and methods used in healthcare tasks of interest, namely, semantic segmentation, boundary detection, and anomaly detection.

Time Series Prediction Tasks. The three tasks we focus on in this work are semantic segmentation, boundary detection, and anomaly detection. Semantic segmentation entails partitioning an input into contiguous segments that represent an object or event, and classifying each segment. This problem has been studied extensively in the context of computer vision, in which it is natural to segment objects in an image and classify them (Hao et al., 2020). It is also a critical task in time series analysis, where the aim is to segment distinct events or phases (Keogh et al., 2004), although it sometimes referred to as just segmentation. Semantic segmentation is typically formulated as a point-wise classification problem, in which each point is classified independently by a trained classification model. Many such point-wise classifiers have been developed, with recent efforts focusing on deep learning models (Perslev et al., 2019a; Gaugel and Reichert, 2023). Boundary prediction is a closely related task that similarly involves partitioning a signal into discrete segments, but in which we do not have semantic labels for each segment. In the medical domain, these tasks have been studied for ECG and breath waveform phase detection, utilizing a range of statistical (Pan and Tompkins, 1985; Noto et al., 2018) and machine learning (Moskalenko et al., 2019; Londhe and Atulkar, 2021; Liang et al., 2022) approaches. Three notable related works perform breath segmentation (Chong et al., 2021; Noto et al., 2018; Chen et al., 2024), but all require a certain set of waveforms as input, which are not always available.

General unsupervised time series anomaly detection has been extensively studied in the literature and there are many diverse approaches to the problem (Lu and Ghorbani,

2008; Salem et al., 2014; Pena et al., 2013). Recently, deep learning based methods, and in particular transformer-based architectures, have garnered significant attention in this problem setting due to their flexibility and strong performance (Wu et al., 2022; Xu et al., 2018a; Gao et al., 2020b). While these methods have had great success on a variety of benchmark datasets, they largely focus exclusively on time series signals without additional unstructured context, which can be of great importance, particularly in specialized domains such as ECG analysis. Many domain-specific methods have also been investigated for time series anomaly detection in medical settings that overlap with the datasets we focus on in this work. In particular, various methods that target anomaly detection in ECG signals have been proposed (Li and Boulanger, 2020; Sivapalan et al., 2022; Alamr and Artoli, 2023). However, these methods tend to be tailored to the specific problem using knowledge of the data and types of anomalies, and thus cannot be readily adapted to other domains.

LLMs for Time Series Analysis. LLMs have been applied to general time series tasks such as forecasting and classification by integrating time series data through prompt augmentation (Xue and Salim, 2023; Gruver et al., 2024), or utilizing pretrained backbones for downstream tasks (Zhou et al., 2024). Liu et al. (2023c) uses explicit domain identification information to allow forecasting strategy adaptation. Our study builds upon the work of Jin et al. (2023) which introduces a "reprogramming" layer to project time series patches onto a pretrained LLM's embedding space, enabling it to make effective use of the raw time series in conjunction with textual context for forecasting. Our study adapts this work for the medical domain in addition to the following methodological contributions: (1) extending it to solve a set of time series tasks that are clinically relevant, (2) improving the way covariates are utilized, and (3) augmenting the text prompt to include patient-specific clinical information.

In the medical domain, LLMs have been used to analyze biomedical signals, particularly in electroencephalogram (EEG) and electrocardiogram (ECG) analyses, for tasks like automated report generation (Duan et al., 2024) and zero-shot disease detection (Wang et al., 2024). Multimodal LLM frameworks have been used to enhance disease risk quantification (Belyaeva et al., 2023) and pattern recognition using wearable data (Liu et al., 2023b). As far as we are aware, our method is the first to apply LLMs to the medical domain for time series tasks. In addition, our method more effectively utilizes time series input data with the aforementioned "reprogramming" layer and our adaptations.

3. Methods

In this study, we develop a multimodal model that can adapt pretrained LLMs for multivariate time series task-solvers. The distinctive feature of this approach is its ability to process both raw time series data and natural language inputs. Our method consists of four core components: (1) prompt generation (Section 3.1), (2) time series embedding (Section 3.2), (3) a pretrained LLM (Section 3.3), and (4) time series task-solvers (Section 3.4), as illustrated in Figure 1, with more details provided in Figure 2. We use dataset, task, patient-specific, and time series information to construct a prompt with relevant context that instructs the LLM to solve the desired task. We split time series into patches and align the patch embeddings with text embeddings from the LLM so that it can utilize them effectively. For alignment, we adopt the patch reprogrammer introduced in Jin et al. (2023), but extend it to incorporate covariates in the framework. We then feed the word and time series embeddings

into state-of-the-art LLMs to analyze the text and time series together. Finally, we take the output embeddings of the LLM and use task-specific projection layers and processing to solve the selected time series analysis task.

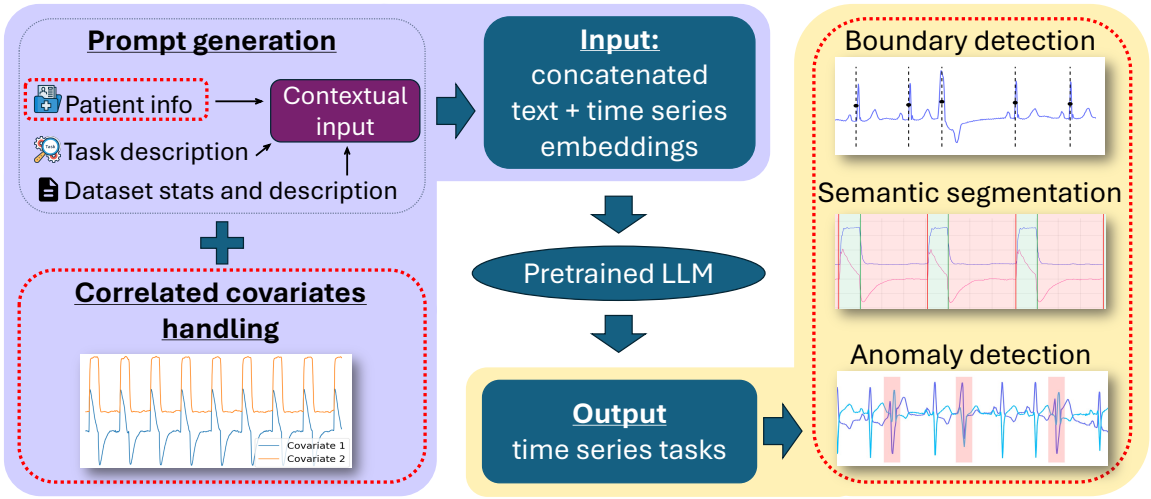

Figure 1: In our proposed framework, the multimodal input consists of contextual input and raw time series data, which are both converted to embeddings. The concatenated embeddings are fed into a pretrained LLM. Output LLM embeddings are then used by task-specific methods to generate predictions. Our contributions are highlighted with red dotted lines.

The three tasks we focus on in this work are anomaly detection, boundary detection, and semantic segmentation. Anomaly detection involves determining which periods of the input deviate from normal behavior. Boundary detection requires finding points that split a time series up into periods that represent a single type of event, such as a heartbeat or breath. Semantic segmentation is similar to boundary detection, but also determines a classification for each sequence, which is useful when there are multiple distinct types of events (e.g., ventilator-delivered breaths vs spontaneous breathing) or distinct event phases (e.g., inspiration and expiration) that are important to capture. These methods are further described in Section 3.4.

3.1. Prompt Generation

We aim to improve the performance of our task solvers by including relevant information that is not captured in the time series or other structured data such as clinical notes, diagnostic reports, and medical history, by capturing it as unstructured natural language and using it to inform time series predictions. Therefore, it is critical that we provide the LLM with useful context and an effective prompt to maximize the power of its language modeling abilities. There are four main components of the context that we provide the LLM: dataset description, task description, a summary of dataset statistics and detailed patient-specific information (see Figure 1). To leverage the LLM's pretraining, we encode all static information as text, rather than developing separate encoding mechanisms per data modality.

The dataset description provides the LLM with the relevant background information to explain the domain and context. It informs the LLM of what the problem setting is, what

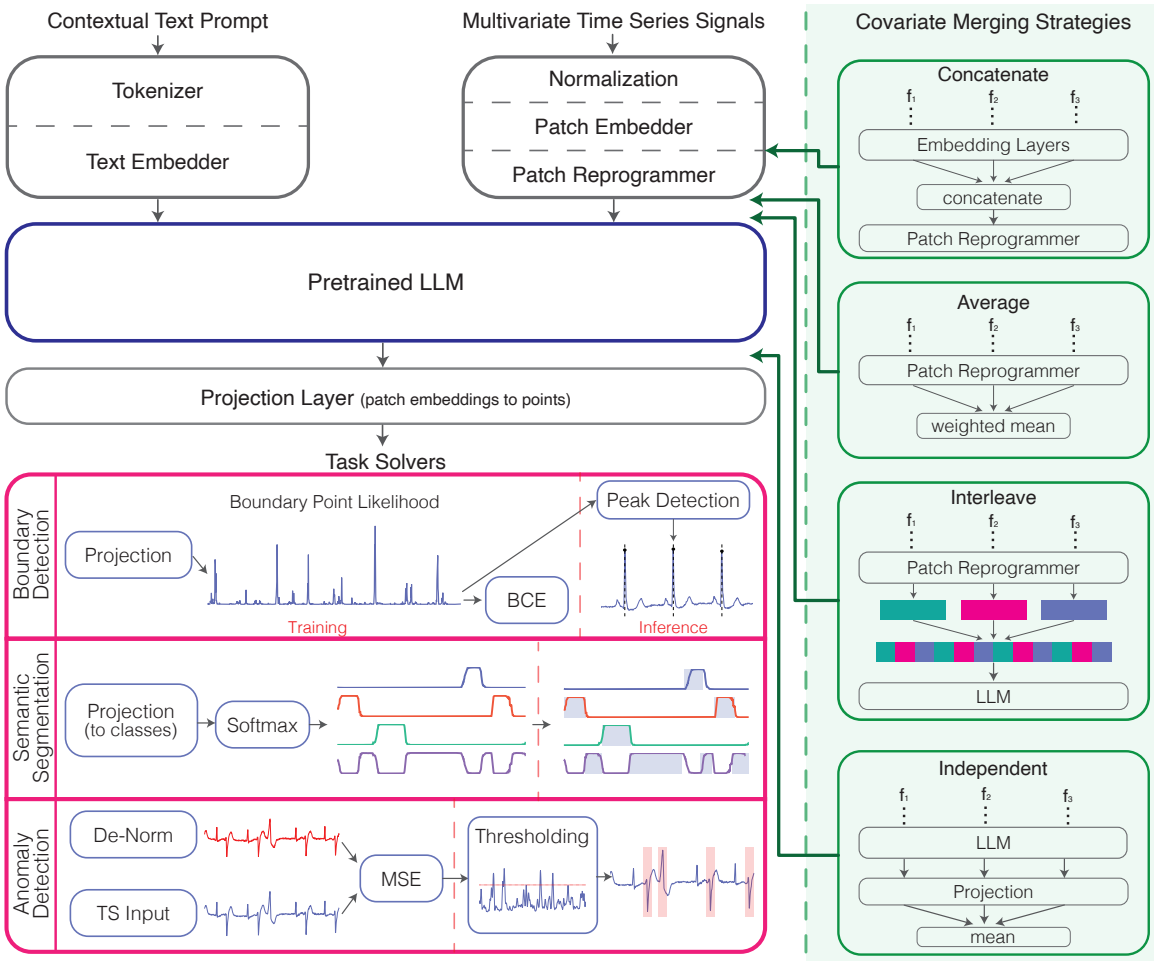

Figure 2: A schematic overview of our proposed methodology. Time series and corresponding textual context are embedded separately and concatenated. While text passes through standard tokenization and embedding, time series is patched and transformed into embeddings, which are aligned to text embeddings using a patch reprogrammer layer. Covariate information is merged using one of our proposed strategies, and embeddings are fed into the LLM, and through a projection layer to produce raw predictions. These predictions are transformed to solve one of our selected analysis tasks using task-specific layers and processing. The covariate strategies (green) and task solvers (pink) are our primary methodological contributions. All plot examples in the task solvers section are real outputs of our model on the datasets used in this work.

modality of data each feature represents, and how the data was collected. This information allows the model to apply any knowledge about the task and data that it may have learned in the pretraining phase. Specific prompts for each dataset can be found in Table 8. Task instructions, important for instruction-tuned LLMs, guide the model's expected output (e.g., "Identify the change points in the past 256 steps of data to segment the sequence."). Some additional summary statistics (e.g. minimum and overall signal trend) are included for low-frequency complementary signals (less than 1 Hz) and encoded as text.

Further information about the specific patient that the time series represents can be highly relevant to the predictions. Large amounts of medical data are stored as unstructured

text, including clinician notes and other reports, which may provide important context about conditions or diagnoses that affect the time series inputs and outputs. The patient-specific information included in the prompt depends on what was available for each dataset, but generally contains patient demographics, administered medications, diagnoses, clinical notes, and additional medical history. To standardize the formatting, particularly for structured demographic information, patient-level data was encoded as JSON, which also improved the model's performance.

The dataset description, patient-level information, summary statistics, and task instruction were concatenated (in that order) into one string and sent through the pretrained LLM's tokenizer and embedding layer to produce token embeddings. These are then concatenated with the time series patch embeddings, which are discussed in Section 3.2, to form the input to the LLM. We also evaluated the contribution of each component in Table 6.

3.2. Time Series Encoding

The main barrier to effectively using pretrained LLMs for time series analysis is encoding time series data in such a way that a pretrained LLM can make good use of it. Tokenization of time series data encoded as text maps 1-3 digits at a time to a discrete embedding, which requires either very low numerical precision or splitting numbers into multiple tokens. This inefficient method results in suboptimal encodings. Our proposed patch embedding method overcomes these limitations by using multiple data points (typically 16 per patch) at full precision, which are then transformed into continuous embeddings. We train a time series encoder for a pretrained, frozen LLM, to learn embeddings of patches of raw time series signals. Thus, the model can utilize multimodal text and time series inputs without converting signals to text, and without computationally expensive fine-tuning of the LLM.

Patching the input signals allows us to construct tokens with semantic resolution more similar to that of natural language. We first take the input signal $x_{1:T} \in \mathbb{R}^T$ and normalize it using Reversible Instance Normalization (RevIN) (Kim et al., 2021) to account for distribution shift over time. From the normalized signal we construct a set of patch vectors $\{x_{t:t+l-1} : i \in P, t = is + 1\}$, where $P = \{i \in \mathbb{Z} : 0 \leq i \leq \frac{T-l}{s}\}$ is the set of patch indices, l is the patch length, and s is the stride. Each patch is then transformed into an embedding using a linear projection layer.

However, in order for a pretrained LLM to make effective use of these embeddings without extensive fine-tuning, the embeddings must be aligned with the distribution of pretrained token embeddings. It is a well understood problem in ML that models often perform poorly on a target domain if there is a significant gap between the target data distribution and the training distribution (Ben-David et al., 2010). Therefore, the LLM backbone will likely not perform well in our model if there is a significant gap between the distribution of text embeddings and time series embeddings. This problem is recognized as a core challenge in multimodal ML (Gao et al., 2020a), and has received significant attention recently for vision-language models.

To address this, we utilize the patch reprogramming layer introduced in Jin et al. (2023) to perform this alignment. The patch reprogrammer performs cross attention between patches and a set of (transformed) token embeddings from the LLM. The result is a representation of each patch as a linear combination of pretrained token embeddings. This is a powerful

inductive bias that improves the ability of the encoder to construct embeddings which match the expected inputs to the LLM.

While Jin et al. (2023) develop a time series forecasting method using the patch reprogrammer, their method treats each covariate independently, so no information mixing between covariates can occur. While this may be suitable for some applications, there are many applications in medicine for which interaction effects are very relevant. For example, ventilator waveform anomalies can accompany changes in vital signs such as increased heart rate or decreased oxygen saturation and increase a clinician's concern for clinically significant dyssynchrony. We therefore introduce a set of methods for incorporating covariate features into the model. There are many possible ways of incorporating these features, each with complex tradeoffs. We propose four approaches: concatenating the covariate signals for each patch, averaging the patch embeddings across covariates, interleaving the embeddings, and averaging the final predictions.

In a multivariate setting, we have a signal $x_{1:T,1:C} \in \mathbb{R}^{T \times C}$ with patches $\{x_{t:t+l-1,c} : c \in \mathbb{N}_{\leq C}, i \in P, t = is + 1\}$. The 'concatenation' strategy passes each covariate signal through the normalization layer and initial embedding layer independently, and concatenates the patch embeddings before the reprogramming layer, forming combined patch embeddings $\phi_t \in \mathbb{R}^{d_p \times C} \forall t \in P$, where d_p is the dimension of each univariate patch embedding. The 'averaging' strategy keeps the embeddings of each covariate independent until after the patch reprogrammer, at which point it computes a weighted average $\phi'_t = \sum_{c \in \mathbb{N}_{\leq C}} w_c \phi_{t,c}$ using learned weights $w_c \in \mathbb{R}_{\geq 0}$ such that $\sum_c w_c = 1$. The 'interleave' strategy, instead of merging the embeddings, interleaves them in the input to the LLM, constructing a sequence $(\phi_{1,1}, ..., \phi_{1,C}, \phi_{2,1}, ..., \phi_{2,C}, ..., \phi_{N,1}, ..., \phi_{N,C})$. The information from each covariate is therefore mixed in the LLM, and combined in a final projection layer of the model. Finally, the 'independent' strategy treats each covariate independently throughout the model, and averages the final outputs. Visualizations of these methods are provided in Figure 2, and a full comparison in Section 6.

3.3. LLM Backbone

The core component of our model that enables it to leverage unstructured text data is a pretrained LLM. Rather than training our own domain-specific LLM, we make use of state-of-the-art publicly available pretrained foundation models. Previous studies have demonstrated that models trained on vast swaths of general data can significantly outperform specialized models (Nori et al., 2023), so we aim to harness their capabilities for the medical time series domain. In order to preserve the language understanding abilities gained through pretraining and minimize computational requirements, we use the LLM in a frozen state, without fine-tuning its parameters. We do not limit our framework to a particular LLM, instead it can be used with any current LLM architecture, as further explored in appendix B.1.

The input to the frozen pretrained LLM is constructed by concatenating the token embeddings produced by the LLM's embedding layer for the textual prompt with the patch embeddings produced by the time series encoder. This allows the model to utilize the semantic information from both the time series and the unstructured metadata in its predictions. After passing the inputs through the model, we extract the token embeddings produced by the final layer.

Since decoder-only transformers are sequence-to-sequence models, the LLM backbone computes an output token for each input token, including both text tokens and time series patches. While the LLM uses the text tokens to generate outputs for each time series patches, the model's purpose is to make predictions only for the time series patches, allowing the outputs corresponding to the text tokens to be safely ignored. The output embeddings corresponding to the time series patches are concatenated and the resulting vector is fed into a linear projection layer, which maps the concatenated embedding vector directly to the output time series points, to make the final predictions for the selected time series analysis problem.

3.4. Task Solvers

3.4.1. Semantic Segmentation

Semantic segmentation is the task of partitioning a time series into discrete segments and assigning a classification to each segment. This can be used to identify each phase of an event (such as inspiratory and expiratory phases of a breath), or pick out specific types of events (such as different types of heartbeat arrhythmias). While this task can be solved through a process of boundary detection then segment classification, these two sub-tasks are inter-related – in particular, the classification of a segment can influence the optimal boundary selection. Therefore, we instead predict a class for each point in the input time series. Point-wise classifications can then be assembled into classified segments by grouping contiguous sets of points with the same predicted label. This approach solves the problem in a single pass, combining the segmentation and classification sub-tasks.

To predict the label for each point, we first predict the likelihood of each class for each point. In the projection layer following the LLM, we map the set of final patch embeddings from the LLM to a matrix of size $N \times C$, where N is the number of time points, and C is the number of classes. We then apply the SOFTMAX function to each row, resulting in a class likelihood vector for each point. We use cross-entropy loss for training, and take the ARGMAX over the likelihood vectors to compute the predicted label for each point during testing. In settings where there are only two classes, it is not necessary to predict the likelihood of both as they must sum to 1. Therefore, we predict only a single value for each point, and use the SIGMOID function to map it to the likelihood of one class. The likelihood is then thresholded at 0.5 to determine the predicted class for every point. In this case, the model is trained with the binary cross entropy loss function.

The Semantic Segmentation pipeline in Figure 2 shows how patch embeddings are projected to multiple class values per time point. After applying the SOFTMAX function, we compute the class label for each point, and at inference, consecutive points with the same class label are combined to form segments.

3.4.2. Boundary Detection

Boundary detection involves partitioning a time series into discrete segments by predicting the boundaries of each segment. It differs from semantic segmentation primarily in that segments are not assigned classifications, and each event that is segmented may be semantically identical to its neighbors. While there is less information to predict in boundary detection, it is typically more challenging than semantic segmentation because it is difficult to delineate

between consecutive periods that may be very similar. MedTsLLM performs boundary detection by classifying each boundary point.

In our boundary point classification method, we treat boundary detection as a binary classification problem, in which every point is either a boundary point between segments or not. For each point in the time series, we predict a single value representing the likelihood of the point being a segment boundary. The model is trained using binary cross entropy (BCE) loss. While the labels are extremely unbalanced, which often poses significant challenges for classifiers, we have found that this method performs well in practice, which we attribute to our approach for identifying boundaries from the raw likelihood scores. Rather than threshold these scores at 0.5, or some other fixed level, we apply an algorithm that finds local maxima (Virtanen et al., 2020) of the score signal, which correspond to the most likely boundary points. We enforce a constraint that no pair of selected points is closer than some distance threshold parameter, which ensures that only one point is selected for each region of elevated scores. An approximately optimal distance parameter can be found using gradient-free optimization methods that maximize the target evaluation metric over the training data, or validation data if available. We also utilize the 10^{th} percentile of segment lengths in the training set as a simple but effective heuristic that is much faster to compute. Figure 2's Boundary Detection section illustrates how after projecting patch embeddings to points representing likelihood scores, the model is trained using BCE to compare the scores against the true binary class labels. At the inference stage, using the described constrained peak-finding algorithm, we identify the most likely peaks in the likelihood scores that ultimately correspond to boundary points.

3.4.3. Anomaly Detection

The final time series analysis task we perform is unsupervised anomaly detection. In this setting we assume we have an unlabeled training set of "normal" periods, and we aim to detect anomalous periods that deviate from normal in unseen data. We utilize the standard approach to this task in the machine learning literature, which consists of training a model to reconstruct the normal input signals, and during inference, marking points in the input as anomalous if they deviate from the predicted signal by more than some threshold. This can be an effective approach as the reconstruction model is only trained to predict normal sequences, so significant deviations from its predictions are likely to be anomalies. More specifically, we train our model to take an input signal and output a signal as similar as possible by minimizing the mean square error (MSE). During testing, the model attempts to reconstruct an input signal, and we compute an anomaly score for each point, which is the MSE between the inputs and predictions, normalized across features. We set a threshold using the frequency of anomalies in each dataset and the distribution of scores, and predict that any points with score above this threshold are anomalous. The Anomaly Detection section in Figure 2 illustrates how the MSE between the denormalized, predicted (reconstructed) signal and the original time series signal is thresholded to identify periods of anomalous points.

4. Datasets

We showcase our model's applicability on multiple datasets spanning two domains: ECGs and respiratory signals. The ventilator data specifically is internally collected with patient consent and approved by the relevant institutional review board. We also discuss the processing of publicly available datasets, extracted from Physionet (Goldberger et al., 2000). Additional summary statistics can be found in Tables 9 to 11.

Table 1: Overview of the datasets used in this study. These statistics pertain to our processed versions of the datasets that we use for analysis rather than the raw data. Only semantic segmentation datasets have classifications.

Dataset	Category	Task	Features	Classes	Time points
Ventilator	Respiration	Semantic segmentation	2	2	988,217
LUDB	ECG	Semantic segmentation	1	4	8,771,395
BIDMC	Respiration	Boundary detection	3	–	3,180,053
MIT-BIH	ECG	Boundary detection	2	–	9,027,800
MIT-BIH Arrhythmia	ECG	Anomaly detection	2	–	7,447,935

4.1. Ventilator Waveforms

To study mechanical ventilation for pediatric patients, we have, with IRB approval, internally collected a dataset of ventilator waveforms and relevant clinical information from N=17 patients from the Pediatric Intensive Care Unit at Johns Hopkins All Children's Hospital from July 2020 to August 2021. The dataset consists of over 1,700 hours of EHR data (e.g. patient demographics, medication administration, etc.) and physiologic time series data (e.g. numeric values and waveforms from GE physiologic monitors and Draeger ventilators). 10 30-minute clips of ventilator pressure and flow waveforms, extracted from a curated subset of 5 patients on pressure-control synchronized intermittent mandatory ventilation, were segmented into inspiratory and expiratory periods. An expert clinician annotated 60% of the waveforms using ECG lead II as a reference, while a trained intern annotated the remaining 40%, which was then checked by the supervising clinician. Small segments that could not be cleanly segmented following expert guidelines were removed, resulting in a total of 7,344 identified breaths.

The dataset includes 7 clips from periods with a ventilator-measured triggered rate of at most 1 (stable, ventilator-delivered breaths) and 3 clips with rates between 2 and 5 (patient-triggered, often noisier breaths), providing a physiologically diverse dataset representative of different patient states on ventilatory support. To facilitate waveform analysis, the prompt part of the dataset includes clip-specific information (e.g. statistics on both ventilator-derived signals like respiratory rate and other signals like heart rate, and ventilator settings), and patient-specific information (e.g. age, gender, medications, etc.).

4.2. Publicly Available Datasets

Lobachevsky University Electrocardiography Database (LUDB). LUDB, a public ECG delineation dataset (Kalyakulina et al., 2020, 2021), contains 200 10-second-long 12-lead ECG signals sampled at 500 Hz from 200 healthy volunteers and patients with various

cardiovascular diseases. Two cardiologists annotated each lead with P wave, T wave, and QRS complex boundaries and peaks for each beat, along with patient diagnoses. The dataset also includes patient information such as sex, heart rhythms, and conduction abnormalities. Classifying every point into one of the three classes (P, T, and QRS), or a fourth "unlabeled" class, is a semantic segmentation task. Each ECG lead is considered a separate univariate time series due to independent annotation. 80% of patients were randomly selected for the training set, and the rest were used for testing.

BIDMC PPG and Respiration The BIDMC PPG and Respiration dataset (Pimentel et al., 2017), collected from critically-ill patients as part of the MIMIC II matched waveform database (Lee et al., 2011), consists of 53 8-minute recordings of various physiological signals (sampled at 125 Hz) and physiological parameters (sampled at 1 Hz). The analysis focuses on signals consistently available across all patients: the impedance respiratory signal, plethysmograph, and ECG lead II. Averages of each physiological parameter such as heart rate are computed for context, along with patient age and sex. Two annotators segmented the data into individual breaths using the impedance respiratory signal, with labels consisting of boundary points between consecutive breaths. For simplicity, only the first annotator's labels are utilized. The dataset is partitioned into training (85% of patients) and test (15% of patients) sets.

MIT-BIH The MIT-BIH dataset (Moody and Mark, 2001) consists of 48 30-minute, 2-channel ambulatory ECG recordings from 47 patients. For consistency, we include only patients with MLII and V1 ECG channels and classify beat annotations as either normal or abnormal, with any label other than normal (e.g., left bundle branch block beat) considered abnormal. The dataset also contains patient information, including age, medications, and annotator notes about identified artifacts in the waveforms. Following Ohhwan et al. (2018), we downsample the signals from 360Hz to 125Hz to reduce the computational demands without compromising relevant information.

We solve two tasks on this dataset: boundary detection for segmenting beats and anomaly detection for identifying arrhythmias, with different splitting and label processing strategies that reflect the particulars of each task. For boundary prediction, labels correspond to the annotated beat peak points, with no distinction between normal and abnormal beats. Patients are partitioned randomly between training (80%) and testing (20%) subsets. For anomaly detection, we select patients with the least number of anomalies for training, and those with the most anomalies for testing, maintaining an 80/20 split. Patients with all abnormal ECG beats were excluded. Each abnormal annotation is expanded to fill the window of 150 ms before and after the annotation, following the American Medical Instruments standard (Association for the Advancement of Medical Instrumentation, 1999).

Non-Medical Datasets As we only have one medical time series dataset for anomaly detection, to demonstrate our method's ability to perform across datasets, we include two additional non-medical datasets frequently used for benchmarking time series anomaly detection methods: the Mars Science Laboratory (MSL) dataset, a public NASA dataset containing expert-labeled telemetry anomaly data (Hundman et al., 2018), and the Pooled Server Metrics (PSM) dataset, which contains internally collected data from multiple application server nodes at eBay (Abdulaal et al., 2021).

5. Results

In this section, we demonstrate the potential of MedTsLLM to solve semantic segmentation, boundary detection, and anomaly detection on our selected datasets (see Section 4), and evaluate its performance relative to other state-of-the-art methods based on metrics and experimental setup defined in Section 5.1. We evaluate our model on various task-solvers in Sections 5.2 to 5.4, and perform ablation studies on prompting and covariate strategies in Section 5.5.

5.1. Evaluation Approach

Metrics. We utilize task-specific metrics to evaluate the performance of our framework. For semantic segmentation, we use the mean segment-wise Intersection over Union (mIoU) and point-wise F1 score, which are commonly used to assess the quality of segmentation predictions in various domains. For anomaly detection, we report the F1 score and Area Under the Receiver Operating Characteristic curve (AUROC), as they are standard metrics for binary classification problems that provide a balanced measure of performance. We compute these metrics period-wise, as opposed to point-wise, using standard procedure employed in benchmarking recent unsupervised anomaly detection methods (Wu et al., 2022; Xu et al., 2018b). To evaluate boundary detection performance, we report the segment-wise mIoU and the accuracy of predicted segments with at least 0.75 IoU overlap with ground truth segments. We also measure the point-wise mean absolute error (MAE) between predicted and actual boundary points. While these are not standard metrics for time series boundary detection, we believe that the segment-wise mIoU is most relevant to downstream applications, while point-wise MAE provides a more granular measure of boundary localization performance.

Baseline methods. We primarily compare the performance of our model against a variety of competitive models for time series analysis tasks that fall into three categories: LLM-based models, other general deep learning models, and domain-specific methods. We conduct a separate comparison with traditional time series analysis methods described in Tables 13 to 15 in the appendix. The LLM-based approach, introduced by Zhou et al. (2024) (GPT4TS), fine-tunes selected layers of GPT-2 on time series patches, and has achieved competitive performance on benchmark datasets. We select three deep learning models for general time series analysis: PatchTST (Nie et al., 2022), TimesNet (Wu et al., 2022), and FEDformer (Zhou et al., 2022). These three models were selected because they were consistently top performers in recent works that evaluate deep learning models on anomaly detection across a range of benchmark datasets. While these LLM and deep learning methods can be used for anomaly detection without modification, none of them natively support semantic segmentation or boundary prediction. We therefore adapt our approach to these tasks for each model. Additionally, we include two domain-specific methods that were developed to solve particular medical time series semantic segmentation tasks. Ventiliser (Chong et al., 2021) is an algorithm designed to segment and classify breathing phases based on ventilator pressure and flow signals. We utilize this method as a baseline for semantic segmentation on our ventilator dataset. On the LUDB dataset, we compare against Perslev et al. (2019b),

which develops a U-Net inspired convolutional neural network to segment ECGs into onsets and offsets of the P and T waves and QRS complexes.

Experiment details. Throughout Sections 5.2 to 5.4, we employ a standardized approach across models for evaluation. Each model is trained for 10 epochs using identical training and task-solver parameters. In these experiments, MedTsLLM uses LLama 2 (7b) (Touvron et al., 2023) as the backbone LLM, with dataset and task prompts, and the "concatenate" covariate strategy. Further details and information about the implementation and code can be found in appendix C.

5.2. Semantic Segmentation

Table 2 presents the results of a semantic segmentation task performed by different models on the Ventilator and LUDB datasets. MedTsLLM achieves the highest F1 scores and IOU values on both datasets, indicating its effectiveness for semantic segmentation. While all models except the domain-specific model perform exceptionally well on the Ventilator dataset, the LUDB dataset appears to be more challenging, with a wider range of performance across the models. This suggests that deep learning models, particularly MedTsLLM and PatchTST, better capture the complex patterns and features present in the LUDB dataset.

Table 2: Semantic segmentation results.

Model	Ventilator		LUDB	
	F1	IoU	F1	IoU
MedTsLLM	**98.92**	**97.86**	**89.89**	**81.73**
GPT4TS	98.81	97.65	78.92	65.78
TimesNet	98.55	97.14	76.24	62.44
PatchTST	98.72	97.46	89.54	81.31
FEDformer	98.66	97.35	74.97	62.62
Domain-specifc	89.18	80.47	40.12	33.59

5.3. Boundary Detection

Table 3 presents the results of boundary prediction performed by different models on two datasets: BIDMC and MIT-BIH. MedTsLLM emerges as the best-performing model for boundary detection on both datasets, demonstrating its strong segmentation capabilities across different domains. PatchTST and TimesNet also show strong performance, consistently ranking among the top three models, while FEDformer performs well on the BIDMC dataset, ranking second, but falls behind TimesNet on the MIT-BIH dataset. GPT4TS has the lowest performance on both datasets, suggesting that it may not be as well-suited for these specific segmentation tasks, despite being an LLM-based method.

5.4. Anomaly Detection

The results in Table 4 suggest that MedTsLLM's consistent top performance across all three datasets indicates its robustness and superior ability to detect anomalies in different types of time series data. The MIT-BIH dataset proves to be the most challenging, with models exhibiting varying levels of performance. The mixed results of GPT4TS and FEDformer

Table 3: BIDMC and MIT-BIH boundary detection results.

Datasets	Model	mIoU	Accuracy @ 0.75 IoU	Change Point MAE	Accuracy (50 pts)
BIDMC	MedTsLLM	**0.87**	**0.84**	**32.59**	**0.84**
	GPT4TS	0.64	0.29	95.13	0.32
	TimesNet	0.71	0.47	72.94	0.53
	PatchTST	0.75	0.69	65.24	0.59
	FEDformer	0.85	0.82	32.95	**0.84**
MIT-BIH	MedTsLLM	**0.89**	**0.90**	**6.40**	**0.98**
	GPT4TS	0.64	0.34	25.86	0.82
	TimesNet	0.86	0.84	7.19	0.97
	PatchTST	0.87	0.87	7.90	0.97
	FEDformer	0.84	0.77	10.01	**0.98**

across datasets indicate that their architectures or training approaches may not be as well-suited for anomaly detection tasks compared to MedTsLLM, PatchTST, and TimesNet.

Table 4: Anomaly detection results.

Model	PSM		MSL		MIT-BIH	
	F1	AUROC	F1	AUROC	F1	AUROC
MedTsLLM	**97.31**	**98.20**	**88.00**	**90.95**	**94.70**	**98.52**
GPT4TS	90.23	91.40	72.51	82.29	72.19	80.83
TimesNet	89.69	90.92	81.80	87.13	88.29	92.20
PatchTST	95.12	95.60	78.65	84.97	89.53	93.70
FEDformer	90.04	90.94	82.22	87.15	51.23	67.21

5.5. Ablation Studies

We perform ablation studies to demonstrate the effects of changing the (1) covariates and (2) prompting strategies on model performance. Table 5 shows the performance of using different covariates on two tasks: segmentation (BIDMC) and anomaly detection (MIT-BIH). Results indicate that across both datasets and tasks, interleaving or concatenating covariates leads to the best performance. Table 6 shows how different ways of handling the prompt affect performance. The strategies with the word 'only' in the label refer to only keeping that component in the prompt. Results show that using patient-specific information to contextualize the time series leads to the best gains in performance. We explore the implications of these results in the Discussion section below.

Table 5: Results of our ablation study on covariate strategy.

Strategy	BIDMC		MIT-BIH (Anomalies)	
	IoU	MAE	F1	AUROC
Concatenate	**86.00**	**37.53**	94.46	96.75
Interleave	84.40	42.91	**95.75**	**97.69**
Average (weighted)	83.96	42.69	92.65	95.87
Average (unweighted)	83.74	43.88	82.25	90.68
Independent	83.99	43.52	88.36	93.92

Table 6: Results of our ablation study on prompting strategies.

Strategy	LUDB		MIT-BIH (Boundaries)	
	IoU	F1	IoU	MAE
No prompt	77.90	87.40	55.61	32.51
Dataset only	80.71	89.21	60.31	22.10
Task only	80.02	88.79	92.17	4.31
Patient only	**80.82**	**89.31**	**92.18**	**4.20**
Stats only	80.75	89.23	56.05	31.11
All	79.50	88.41	57.07	28.76

6. Discussion

MedTsLLM consistently outperforms state-of-the-art baselines, showcasing the potential of leveraging LLMs for analyzing complex physiological signals. In this section, we explore the technical and clinical implications of our methodology.

6.1. Technical Implications

Domain-agnostic approach. Our study shows that our model can perform well across different medical applications, including respiration and ECG, and for different time series tasks. This provides evidence of the generalizability of LLM-based frameworks and suggests that LLMs may be useful for a diverse range of tasks in healthcare. In addition, using LLMs opens up opportunities for future work to make use of currently underutilized sources of rich medical data, like EHR, which contain semi-structured, heterogeneous data.

Covariate strategies. Integrating information across interrelated signals is critical in multivariate time series analysis, particularly in the medical domain, where multiple physiologic signals can have complex interactions that should drive predictions. Several recent deep learning methods for general time series forecasting have challenged this outlook (Nie et al., 2022; Jin et al., 2023), including TimeLLM, which we derive our time series encoder from, and have found success on standard benchmark datasets by treating covariates independently. However, our ablation studies convincingly demonstrate that across a range of medical tasks, our method performs significantly better when utilizing multivariate information. Furthermore, covariate handling strategies that allow the model to best integrate information across covariates, in particular the interleave strategy, also perform best.

Table 7 summarizes the pros and cons of each strategy. While task performance is ultimately the most consequential consideration in selecting a strategy, other factors may influence the optimal method for a particular dataset or task. In particular, interleaving covariate tokens in the LLM input allows for flexibility in handling missing information, different frequencies, or irregular time steps, and allows the LLM to attend to each covariate patch and determine how to mix information across them. However, it multiplies the size of the input sequence to the LLM by the number of covariates included, so it does not scale well to higher dimensional datasets. Instead, the embedding averaging strategy can be used which requires limited additional memory or computational power with increasing covariates, and retains the flexibility, but does not perform as well. A more balanced approach is concatenating covariate patches, which scales well and achieves strong performance.

Table 7: Comparision of different covariate strategies.

Property	Average	Concatenate	Interleave	Independent
Memory scaling	★★★	★★	★	★★
Information retention	★	★★	★★★	★★
Data flexibility	★★★	★	★★★	★★
Performance	★★	★★★	★★★	★

Prompting strategy. Recent publicly available LLMs have been trained on massive a corpus of information collected from the internet, giving them "knowledge" of medical conditions, time series analysis, specific medical time series tasks, and other valuable information for our problem (Singhal et al., 2023). Effective prompting of the LLM is essential to ensure that it can utilize this knowledge and apply it to its predictions. We therefore have constructed prompting strategies to provide the necessary context and instructions to maximize the performance of LLMs on our tasks. Information about the general domain of the problem and the specific dataset are included to ground the LLM's predictions in this critical context. Patient-specific information allows for incorporating additional unstructured data for each time series, which is not otherwise accessible for time series models. Statistics about the input signals and other low frequency data in text format help to provide information about the signals in the modality that LLMs are trained to use. Finally, a task instruction tells the LLM what its outputs should be, which can be particularly valuable if using instruction or chat-tuned LLMs. Our prompting ablation studies demonstrate the value of each of these prompt components in improving the performance of MedTsLLM. However, we find that with too many of these prompt components used at once, performance can suffer. We hypothesize that this is likely due to the LLM struggling to determine what information is important when the prompt grows too large, which has been observed in other settings (Liu et al., 2023a), and is a critical problem to address in future work.

6.2. Clinical Implications

Analyzing physiological signals is crucial for clinical decision-making, as insights extracted from them help clinicians better understand patient state in real time. Our three tasks enable different types of analysis that contribute to this understanding. Using ECGs as an example, our model's accurate boundary detection enables heart rate calculation and heart rate variability analysis, providing critical insights into cardiovascular health. Semantic segmentation further delineates heartbeats into P and T waves and the QRS complex, which are essential for diagnosing cardiac disease states. Accurate and timely anomaly detection allows clinicians to monitor a patient's state and make responsive treatment decisions.

Potential in dyssynchrony analysis. We further explore the applicability of our model within the context of mechanical ventilation. Mechanical ventilation can be life-threatening when suboptimal settings result in patient-ventilator dyssynchrony (PVD), which clinicians struggle to detect and manage due to information overload and limited time for observation before needing to intervene. Addressing this challenge requires automated and accurate detection of PVD. Current dyssynchrony research necessitates precise breath segmentation labels obtained from pressure and flow waveforms. Unfortunately, the availability of publicly accessible, labeled, high-frequency ventilator waveform datasets and breath segmentation tools is severely limited compared to other medical domains, such as ECG analysis.

Our work offers valuable contributions in two significant ways. Firstly, with just 5 hours of labeling, our model performs better than clinical segmentation tools, without even enforcing specific breath waveform types. This could encourage other research teams to utilize our tool for building more breath segmentation datasets. Secondly, our model's anomaly detection abilities could be harnessed for PVD detection. Although we could not validate its performance due to the lack of anomaly labels, the potential is promising. One common obstacle faced by researchers in the dyssynchrony space is the extensive resources required for segmenting data into inspiration and expiration phases before labeling PVD. Our model has the potential to streamline this process by aiding in both segmentation and identifying anomalies or dyssynchronies for clinical review.

Leveraging underutilized multimodal data. One of the key factors that contribute to our model's success is its ability to incorporate heterogeneous data modalities, which is particularly useful when dealing with a mixture of time series signals and EHRs. EHRs contain data of various modes, frequencies, temporal resolutions, and distributions, which traditional time series models cannot handle effectively (and hence do not include). In contrast, our natural language prompting strategy can fuse these diverse data types without requiring conversion to standardized, structured format. This enables inclusion of the wide-ranging data extracted from each patient, and our ablation studies 6 confirm that it does significantly improves performance. By incorporating raw time series signals, our model offers an alternative to existing LLMs that process EHR data solely through clinical notes or require inputs to be in a specific format. This flexibility enables our model to leverage the full potential of the diverse data available from each patient, leading to more accurate and comprehensive analyses.

6.3. Limitations and future directions

Despite our promising method and strong results presented in our work, there are several limitations that should be addressed for future research. Firstly, while MedTsLLM demonstrates high performance, interpretability of the model's predictions remains a challenge. Future work should focus on developing methods to explain the model's decisions and provide more transparent insights to clinicians, perhaps through text output. To note, however, our focus is on relatively straightforward, lower-level tasks that are too time-intensive for humans to perform at scale, not decision-making. The high volume of model predictions renders it impractical for clinicians to audit each one. In these situations, we believe model performance is more critical for trustworthiness than interpretability is, and have hence prioritized the former.

Secondly, compared with simpler time series models, our model is more computationally intensive to train. In this study, we have decided to focus on maximizing our method's performance rather than its computational efficiency, especially as ML methods become increasingly prevalent in medicine, and as clinical environments update their infrastructure accordingly. Nonetheless, while training LLM models of various sizes, we discovered that using smaller ones can significantly reduce computational requirements with only a small drop in performance (Table 12). Future work can explore optimizations for LLM inference like quantization and key-value (KV) caching, to be used in resource-constrained environments.

Our current approach involves freezing the LLM backbone and training specific layers. While fine-tuning the LLM backbone on domain-specific medical data could potentially enhance its adaptation to healthcare applications, the computational demands of our method render this impractical. Even with parameter-efficient techniques like LoRA (Hu et al., 2021), the increased cost and complexity are likely to outweigh any performance benefits.

It is important to note that our method does require fine-tuning specific layers for each dataset and task. Future research could explore training an LLM from scratch to inherently comprehend time series data, which might also improve generalizability. However, this approach faces a significant challenge, one we encountered when seeking to expand our method to other medical domains: the scarcity of suitable public clinical datasets, especially outside cardiology and pulmonology. The limited availability of large-scale, appropriately labeled time series datasets in other medical domains has constrained the scope of such endeavors.

One alternate way to further realize the potential of MedTsLLM in real-world clinical settings is incorporating more EHR data and extensive clinical notes to help the LLM build a more detailed profile of each patient. Another future direction would be adding more task functionality to MedTsLLM, such as forecasting, clustering, and classification. In combination, these directions can contribute to the development of more powerful, transparent, and widely applicable tools for clinical decision support and personalized medicine.

6.4. Conclusion

Our work introduces MedTsLLM, a novel approach that leverages the power of large language models for medical time series analysis. By integrating patient-specific contextual information and handling multiple covariates, MedTsLLM outperforms state-of-the-art baselines on critical tasks such as boundary detection, semantic segmentation and anomaly detection. This work represents a significant step towards building general-purpose models that effectively combine insights extracted from multimodal data and knowledge from pretrained LLMs, paving the way for more accurate, actionable, and personalized insights from multi-dimensional physiological signals. Clinically, MedTsLLM has the potential to transform patient monitoring, clinical decision support, and personalized medicine, ultimately improving patient outcomes and advancing the field of healthcare.

References

Ahmed Abdulaal, Zhuanghua Liu, and Tomer Lancewicki. Practical approach to asynchronous multivariate time series anomaly detection and localization. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 2485–2494, 2021.

Abrar Alamr and Abdelmonim Artoli. Unsupervised transformer-based anomaly detection in ecg signals. *Algorithms*, 16(3):152, 2023.

Association for the Advancement of Medical Instrumentation. NSI/AAMI EC57:1998/(R)2008 (Revision of AAMI ECAR:1987). Technical report, 1999.

Anastasiya Belyaeva, Justin Cosentino, Farhad Hormozdiari, Krish Eswaran, Shravya Shetty, Greg Corrado, Andrew Carroll, Cory Y. McLean, and Nicholas A. Furlotte. Multimodal llms for health grounded in individual-specific data, 2023.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010.

Cheng Chen, Zunliang Wang, Chuang Chen, Xuan Wang, and Songqiao Liu. A software tool for anomaly detection and labeling of ventilator waveforms. In Guangzhi Wang, Dezhong Yao, Zhongze Gu, Yi Peng, Shanbao Tong, and Chengyu Liu, editors, *12th Asian-Pacific Conference on Medical and Biological Engineering*, pages 277–283, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-51485-2.

David Chong, Colin J. Morley, and Gusztav Belteki. Computational analysis of neonatal ventilator waveforms and loops. *Pediatric Research*, 89(6):1432–1441, May 2021. ISSN 0031-3998, 1530-0447. doi:10.1038/s41390-020-01301-9. URL https://www.nature.com/articles/s41390-020-01301-9.

Yiqun Duan, Charles Chau, Zhen Wang, Yu-Kai Wang, and Chin-teng Lin. Dewave: Discrete encoding of eeg waves for eeg to text translation. *Advances in Neural Information Processing Systems*, 36, 2024.

Jing Gao, Peng Li, Zhikui Chen, and Jianing Zhang. A survey on deep learning for multimodal data fusion. *Neural Computation*, 32(5):829–864, 2020a.

Jingkun Gao, Xiaomin Song, Qingsong Wen, Pichao Wang, Liang Sun, and Huan Xu. Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks. *arXiv preprint arXiv:2002.09545*, 2020b.

Stefan Gaugel and Manfred Reichert. Prectime: A deep learning architecture for precise time series segmentation in industrial manufacturing operations. *Engineering Applications of Artificial Intelligence*, 122:106078, June 2023. ISSN 0952-1976. doi:10.1016/j.engappai.2023.106078. URL http://dx.doi.org/10.1016/j.engappai.2023.106078.

Akash Ghosh, Arkadeep Acharya, Raghav Jain, Sriparna Saha, Aman Chadha, and Setu Sinha. Clipsyntel: clip and llm synergy for multimodal question summarization in healthcare. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22031–22039, 2024.

Ary L Goldberger, Luis A Nunes Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger Mark, ..., and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. *Circulation [Online]*, 101(23):e215–e220, 2000. URL https://www.physionet.org/physiotools/.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

Shijie Hao, Yuan Zhou, and Yanrong Guo. A brief survey on semantic segmentation with deep learning. *Neurocomputing*, 406:302–321, 2020.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Söderström. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. *CoRR*, abs/1802.04431, 2018. URL http://arxiv.org/abs/1802.04431.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

HyoJe Jung, Yunha Kim, Heejung Choi, Hyeram Seo, Minkyoung Kim, JiYe Han, Gaeun Kee, Seohyun Park, Soyoung Ko, Byeolhee Kim, et al. Enhancing clinical efficiency through llm: Discharge note generation for cardiac patients. *arXiv preprint arXiv:2404.05144*, 2024.

Alena I Kalyakulina, Igor I Yusipov, Viktor A Moskalenko, Alexander V Nikolskiy, Konstantin A Kosonogov, Grigory V Osipov, Nikolai Yu Zolotykh, and Mikhail V Ivanchenko. Ludb: a new open-access validation tool for electrocardiogram delineation algorithms. *IEEE access*, 8:186181–186190, 2020.

Alina Kalyakulina, Ilnar Yusipov, Valentina Moskalenko, Alexander Nikolskiy, Kirill Kosonogov, Nikolay Zolotykh, and Maksim Ivanchenko. Lobachevsky university electrocardiography database (version 1.0.1). PhysioNet, 2021. URL https://doi.org/10.13026/eegm-h675.

Eamonn Keogh, Selina Chu, David Hart, and Michael Pazzani. Segmenting time series: A survey and novel approach. In *Data mining in time series databases*, pages 1–21. World Scientific, 2004.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=cGDAkQo1C0p.

Joon Lee, Daniel J Scott, Mauricio Villarroel, Gari D Clifford, Mohammed Saeed, and Roger G Mark. Open-access mimic-ii database for intensive care research. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 8315–8318. IEEE, 2011.

Hongzu Li and Pierre Boulanger. A survey of heart anomaly detection using ambulatory electrocardiogram (ecg). *Sensors*, 20(5):1461, 2020.

Lingyao Li, Jiayan Zhou, Zhenxiang Gao, Wenyue Hua, Lizhou Fan, Huizi Yu, Loni Hagen, Yonfeng Zhang, Themistocles L Assimes, Libby Hemphill, et al. A scoping review of using large language models (llms) to investigate electronic health records (ehrs). *arXiv preprint arXiv:2405.03066*, 2024.

Xiaohong Liang, Liping Li, Yuanyuan Liu, Dan Chen, Xinpei Wang, Shunbo Hu, Jikuo Wang, Huan Zhang, Chengfa Sun, and Changchun Liu. Ecg_segnet: An ecg delineation model based on the encoder-decoder structure. *Computers in biology and medicine*, 145: 105445, 2022.

Che Liu, Zhongwei Wan, Cheng Ouyang, Anand Shah, Wenjia Bai, and Rossella Arcucci. Zero-shot ecg classification with multimodal learning and test-time clinical knowledge enhancement. *arXiv preprint arXiv:2403.06659*, 2024.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. 2023a. doi:10.48550/arXiv.2307.03172.

Xin Liu, Daniel McDuff, Geza Kovacs, Isaac Galatzer-Levy, Jacob Sunshine, Jiening Zhan, Ming-Zher Poh, Shun Liao, Paolo Di Achille, and Shwetak Patel. Large language models are few-shot health learners, 2023b.

Xu Liu, Junfeng Hu, Yuan Li, Shizhe Diao, Yuxuan Liang, Bryan Hooi, and Roger Zimmermann. Unitime: A language-empowered unified model for cross-domain time series forecasting. *arXiv preprint arXiv:2310.09751*, 2023c.

Aboli N Londhe and Mithilesh Atulkar. Semantic segmentation of ecg waves using hybrid channel-mix convolutional and bidirectional lstm. *Biomedical Signal Processing and Control*, 63:102162, 2021.

Wei Lu and Ali A Ghorbani. Network anomaly detection based on wavelet analysis. *EURASIP Journal on Advances in Signal processing*, 2009:1–16, 2008.

George B Moody and Roger G Mark. The impact of the mit-bih arrhythmia database. *IEEE engineering in medicine and biology magazine*, 20(3):45–50, 2001.

Viktor Moskalenko, Nikolai Zolotykh, and Grigory Osipov. *Deep Learning for ECG Segmentation*, page 246–254. Springer International Publishing, September 2019. ISBN 9783030304256. doi:10.1007/978-3-030-30425-6_29. URL http://dx.doi.org/10.1007/978-3-030-30425-6_29.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *The Eleventh International Conference on Learning Representations*, September 2022. URL https://openreview.net/forum?id=Jbdc0vTOcol.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicoló Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, Renqian Luo, Scott Mayer McKinney, Robert Osazuwa Ness, Hoifung Poon, Tao Qin, Naoto Usuyama, Chris White, and Eric Horvitz. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *ArXiv*, abs/2311.16452, 2023. URL https://api.semanticscholar.org/CorpusID:265466787.

Torben Noto, Guangyu Zhou, Stephan Schuele, Jessica Templer, and Christina Zelano. Automated analysis of breathing waveforms using BreathMetrics: a respiratory signal processing toolbox. *Chemical Senses*, 43(8):583–597, 07 2018. ISSN 0379-864X. doi:10.1093/chemse/bjy045. URL https://doi.org/10.1093/chemse/bjy045.

Kwon Ohhwan, Jeong Jinwoo, Kim Hyung Bin, Kwon In Ho, Park Song Yi, Kim Ji Eun, and Choi Yuri. Electrocardiogram sampling frequency range acceptable for heart rate variability analysis. *Healthc Inform Res*, 24(3):198–206, 2018. doi:10.4258/hir.2018.24.3.198. URL http://e-hir.org/journal/view.php?number=936.

Jiapu Pan and Willis J. Tompkins. A real-time qrs detection algorithm. *IEEE Transactions on Biomedical Engineering*, BME-32(3):230–236, 1985. doi:10.1109/TBME.1985.325532.

Eduardo HM Pena, Marcos VO de Assis, and Mario Lemes Proença. Anomaly detection using forecasting methods arima and hwds. In *2013 32nd international conference of the chilean computer science society (sccc)*, pages 63–66. IEEE, 2013.

Mathias Perslev, Michael Jensen, Sune Darkner, Poul Jø rgen Jennum, and Christian Igel. U-time: A fully convolutional network for time series segmentation applied to sleep staging. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 4415–4426. Curran Associates, Inc., 2019a. URL http://papers.nips.cc/paper/8692-u-time-a-fully-convolutional-network-for-time-series-segmentation-applied-to-sleep-pdf.

Mathias Perslev, Michael Jensen, Sune Darkner, Poul Jø rgen Jennum, and Christian Igel. U-time: A fully convolutional network for time series segmentation applied to sleep staging. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 4415–4426. Curran Associates, Inc., 2019b. URL http://papers.nips.cc/paper/8692-u-time-a-fully-convolutional-network-for-time-series-segmentation-applied-to-sleep-pdf.

Marco A. F. Pimentel, Alistair E. W. Johnson, Peter H. Charlton, Drew Birrenkott, Peter J. Watkinson, Lionel Tarassenko, and David A. Clifton. Toward a robust estimation of respiratory rate from pulse oximeters. *IEEE Transactions on Biomedical Engineering*, 64 (8):1914–1923, 2017. doi:10.1109/TBME.2016.2613124.

Osman Salem, Alexey Guerassimov, Ahmed Mehaoua, Anthony Marcus, and Borko Furht. Anomaly detection in medical wireless sensor networks using svm and linear regression models. *International Journal of E-Health and Medical Communications (IJEHMC)*, 5(1): 20–45, 2014.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

Gawsalyan Sivapalan, Koushik Kumar Nundy, Soumyabrata Dev, Barry Cardiff, and Deepu John. Annet: A lightweight neural network for ecg anomaly detection in iot edge sensors. *IEEE Transactions on Biomedical Circuits and Systems*, 16(1):24–35, 2022.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi:10.1038/s41592-019-0686-2.

Zhongwei Wan, Che Liu, Xin Wang, Chaofan Tao, Hui Shen, Zhenwu Peng, Jie Fu, Rossella Arcucci, Huaxiu Yao, and Mi Zhang. Electrocardiogram instruction tuning for report generation. *arXiv preprint arXiv:2403.04945*, 2024.

Jiaqi Wang, Zhenxi Song, Zhengyu Ma, Xipeng Qiu, Min Zhang, and Zhiguo Zhang. Enhancing eeg-to-text decoding through transferable representations from pre-trained contrastive eeg-text masked autoencoder. *arXiv preprint arXiv:2402.17433*, 2024.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The eleventh international conference on learning representations*, 2022.

Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, et al. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In *Proceedings of the 2018 world wide web conference*, pages 187–196, 2018a.

Haowen Xu, Yang Feng, Jie Chen, Zhaogang Wang, Honglin Qiao, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, and Dan Pei. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web - WWW '18*, WWW '18. ACM Press, 2018b. doi:10.1145/3178876.3185996. URL http://dx.doi.org/10.1145/3178876.3185996.

Hao Xue and Flora D Salim. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 2023.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*, 2023.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pages 27268–27286. PMLR, 2022.

Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36, 2024.

Appendix A. Additional dataset information

Table 8: Descriptions of each dataset that are used for prompting MedTsLLM.

Dataset	Description
Ventilator	The dataset contains time series data of airway pressure and flow rate measurements collected from a mechanical ventilator during the respiratory support of a fully sedated patient. The data is sampled at a frequency of 100 Hz. The airway pressure is measured in cmH_2O and the flow rate is measured in L/min.
LUDB	LUDB is an ECG signal database collected from subjects with various cardiovascular diseases used for ECG delineation. Cardiologists manually annotated boundaries of P, T waves and QRS complexes. Each clip consists of a 10-second signal from a single ECG lead, sampled at 500 Hz.
BIDMC	The BIDMC dataset is a dataset of electrocardiogram (ECG), pulse oximetry, photoplethysmogram (PPG) and impedance pneumography respiratory signals acquired from intensive care patients. Two annotators manually annotated individual breaths in each recording using the impedance respiratory signal.
MIT-BIH	The MIT-BIH Arrhythmia Database contains excerpts of two-channel ambulatory ECG from a mixed population of inpatients and outpatients, digitized at 360 samples per second per channel with 11-bit resolution over a 10 mV range.

Table 9: Semantic segmentation dataset statistics.

Dataset	Health domain	# dimensions	# classes	# Train	# Test	# Train class distribution	# Test class distribution
Ventilator	Respiration	3	2	395,665	69,823	I: 0.267 E: 0.709	I: 0.229 E: 0.771
LUDB	ECG	1 (12 leads fed independently)	4	6,988,275	1,783,120	P: 0.091 N: 0.122 T: 0.199 U: 0.589	P: 0.103 N: 0.127 T: 0.127 U: 0.565

Table 10: Boundary detection dataset statistics.

Dataset	Health domain	# dimensions	# classes	# Train	# Test	Train boundary point ratio	Test boundary point ratio
BIDMC	Respiration	3	2	2,520,042	660,011	0.011389	0.00946
MIT-BIH	ECG	2	2	1,128,475	315,973	0.002296	0.002251

Table 11: Anomaly detection dataset statistics.

Dataset	**Application**	**# dimensions**	**# Training**	**# Test**	**Anomaly ratio**
MSL	Space	55	58,317	73,729	0.105
PSM	Server	26	132,481	87,841	0.278
MIT-BIH Arrhythmia	Health	2	1,354,170	902,780	0.261

Appendix B. Additional Results

B.1. LLM Ablation Study

Table 12: Ablation study results comparing performance across LLM backbones for semantic segmentation on the LUDB dataset.

LLM	**Parameters**	**IoU**
Llama 2 7b Chat	6.7B	81.73
BioMedLM	2.7B	81.25
Mamba 2.8b	2.8B	81.09
GPT2 XL	1.6B	80.50
Llama 2 7b	6.7B	80.25
Mamba 1.4b	1.4B	79.76
GPT2	137M	78.11

B.2. Traditional time series results comparison

Table 13: Comparing semantic segmentation results for MedTsLLM with traditional time series analysis methods on the LUDB dataset.

Method	**IoU**	**F1**	**Description**
MedTsLLM	**97.86**	**98.92**	Our proposed method
Thresholding	89.30	94.34	Predict expiration for points with flow<0.05
KNN	91.47	95.54	Point-wise K-nearest neighbors classifier
HMM	90.48	95.00	Hidden Markov Model

Table 14: Comparing boundary detection results for MedTsLLM with traditional time series analysis methods on the BIDMC dataset.

Method	**mIoU**	**MAE**	**Description**
MedTsLLM	**86.56**	**32.59**	Our proposed method
Peak Detection	74.96	94.67	Segment boundaries are usually at peaks of the RESP signal, so run SciPy's peak detection algorithm.
Template Matching	70.82	81.369	Select 20 template breaths and use dynamic time warping distance to find segments.

Table 15: Comparing anomaly detection results for MedTsLLM with traditional time series analysis methods on the MIT-BIH dataset.

Method	F1	AUROC	Description
MedTsLLM	**94.7**	**98.52**	Our proposed method
Quantile-based	81.11	88.23	Flag points outside specified quantile thresholds (e.g., 5^{th} and 95^{th} percentiles).
Z-score	72.52	82.23	Flag points beyond a set number of standard deviations (z-score) from the mean.
Rolling average	55	69.18	Flag points deviating from the rolling average beyond a set threshold.
FFT-based	52.58	68.02	Threshold reconstruction error from an FFT-based model.

Appendix C. Experimental Setup and Implementation

C.1. Implementation

- Versions: Python 3.10, PyTorch 2.2.1, Transformers 4.39.3
- GPU: A100 80GB
- Pretrained LLMs from HuggingFace