IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 December 2024, accepted 23 January 2025, date of publication 28 January 2025, date of current version 19 February 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3535782

SURVEY

Time-Series Large Language Models: A Systematic Review of State-of-the-Art

SHAMSU ABDULLAHI[1,2], KAMALUDDEEN USMAN DANYARO[1], (Member, IEEE), ABUBAKAR ZAKARI[1,3], IZZATDIN ABDUL AZIZ[1], NOOR AMILA WAN ABDULLAH ZAWAWI[4], AND SHAMSUDDEEN ADAMU[1]

[1]Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Seri Iskandar, Perak 32610, Malaysia
[2]Department of Computer Science, Hassan Usman Katsina Polytechnic, Katsina, Katsina State 820102, Nigeria
[3]Department of Computer Science, Aliko Dangote University of Science and Technology, Wudil, Kano 713101, Nigeria
[4]Civil and Environmental Engineering Department, Universiti Teknologi PETRONAS, Seri Iskandar, Perak 32610, Malaysia

Corresponding author: Shamsu Abdullahi (shamsu_24001499@utp.edu.my)

This work was supported by the Yayasan Universiti Teknologi PETRONAS: Pre-Commercialization Research Grant (YUTP-PRG) through the Grant Title: Assessment of Structural Steel Riser-Guard Load Capacities on the Resultant Impact on Offshore Structures Loading under Grant 015PBC-011.

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

ABSTRACT Large Language Models (LLMs) have transformed Natural Language Processing (NLP) and Software Engineering by fostering innovation, streamlining processes, and enabling data-driven decision-making. Recently, the adoption of LLMs in time-series analysis has catalyzed the emergence of time-series LLMs, a rapidly evolving research area. Existing reviews provide foundational insights into time-series LLMs but lack a comprehensive examination of recent advancements and do not adequately address critical challenges in this domain. This Systematic Literature Review (SLR) bridges these gaps by analysing state-of-the-art contributions in time-series LLMs, focusing on architectural innovations, tokenisation strategies, tasks, datasets, evaluation metrics, and unresolved challenges. Using a rigorous methodology based on PRISMA guidelines, over 700 studies from 2020 to 2024 were reviewed, with 59 relevant studies selected from journals, conferences, and workshops. Key findings reveal advancements in architectures and novel tokenization strategies tailored for temporal data. Forecasting dominates the identified tasks with 79.66% of the selected studies, while classification and anomaly detection remain underexplored. Furthermore, the analysis reveals a strong reliance on datasets from the energy and transportation domains, highlighting the need for more diverse datasets. Despite these advancements, significant challenges persist, including tokenization inefficiencies, prediction hallucinations, and difficulties in modelling long-term dependencies. These issues hinder the robustness, scalability, and adaptability of time-series LLMs across diverse applications. To address these challenges, this SLR outlines a research roadmap emphasizing the improvement of tokenization methods, the development of mechanisms for capturing long-term dependencies, the mitigation of hallucination effects, and the design of scalable, interpretable models for diverse time-series tasks.

INDEX TERMS Time-series, large language models, forecasting, tokenization, time-series LLMs.

I. INTRODUCTION

The increasing complexity of real-world data has made time-series analysis indispensable across various fields [1]. This complexity arises from temporal dynamics and intricate

The associate editor coordinating the review of this manuscript and approving it for publication was Chao Tong.

interdependencies among variables, which pose significant analytical challenges. Consequently, time-series analysis has become critical in domains such as energy, transportation, healthcare, finance, and meteorology [2]. Traditional methods for time-series analysis, including statistical approaches [3] and machine learning techniques [4], have achieved notable success in many applications. However, these methods often struggle with high-dimensional data, missing values, and the need to capture long-term dependencies, all of which are essential for accurate temporal modeling [5].

The advent of Large Language Models (LLMs), built on transformer architectures with billions of parameters and pre-trained on vast text corpora, has revolutionized computational capabilities in Natural Language Processing (NLP) [6], [7]. LLMs' ability to capture long-range dependencies and complex sequential patterns via self-attention mechanisms [8] has sparked interest in their application to time-series analysis. This interest has driven the development of time-series LLMs, which apply these models to tackle inherent challenges in time-series data, including nonstationarity and variable interdependencies [9], [10].

Time-series LLMs offer a distinct advantage by leveraging the capabilities of pre-trained LLMs or being developed directly from scratch on temporal data. These capabilities enhance robustness, significantly improving prediction accuracy and the interpretation of sequential patterns [11]. Moreover, the limited availability of specialised time-series datasets has made fine-tuning pre-trained LLMs a widely adopted strategy, enabling these models to perform exceptionally well in tasks such as forecasting, classification, anomaly detection, and imputation [12], [13].

Over the last five years, there has been an exponential rise in research exploring time-series LLMs, producing notable advancements across various tasks. Despite the publication of several surveys and reviews [11], [15], [16], [17], [18], no comprehensive Systematic Literature Review (SLR) has been conducted to thoroughly investigate the innovations and challenges specific to time-series LLMs. Such a review is crucial for consolidating existing knowledge and identifying gaps that can drive future research. This study addresses this gap by conducting an evidence-based SLR that systematically identifies, categorizes, and analyzes studies on time-series LLMs published between 2020 and 2024. The review provides an in-depth synthesis of their contributions, architectural advancements, tokenisation strategies, datasets, evaluation metrics, and the challenges faced in this domain. It also proposes a novel classification scheme to organize the research landscape and highlights future directions for the field.

The study contributions are as follows:

- A detailed and critical systematic review of time-series LLMs, offering insights into state-of-the-art developments.
- The development of a novel classification scheme to categorize time-series LLM research efforts effectively.
- A comprehensive synthesis of contributions, architectures, tokenization strategies, tasks, datasets, and evaluation metrics in the field.
- An identification of key challenges and potential research directions to guide future work.

The remainder of the paper is organized as follows: Section II details the background and related work, research methodology is described in Section III; Section IV presents a thorough analysis of the results; Section V discusses the findings; Section VI outlines potential research challenges; and Section VII concludes the study.

II. STUDY BACKGROUND AND LITERATURE REVIEW

This section presents an overview of time-series concepts, the emergence of time-series LLMs, and a review of related work. It also identifies research gaps in the field of time-series LLMs (see Table 1) and discusses potential threats to validity.

A. TIME-SERIES ANALYSIS

Time-series data is a sequence of data points recorded at regular intervals, represented as $X = \{x_1, x_2, \cdots, x_n\} \in \mathbb{R}^{n \times D}$, where X is the time-series dataset, $x_n \in \mathbb{R}^D$ is the data point at time step n, and D is the dimensionality of each data point. For univariate time series, $D = 1$, while for multivariate time series, $D > 1$, denoted as:

$$X = \left\{ \left(x_1^1, x_1^2, \ldots, x_1^n\right), \left(x_2^1, x_2^2, \ldots, x_2^n\right), \ldots, \left(x_n^1, x_n^2, \ldots, x_n^n\right) \right\},$$

where $X \in \mathbb{R}^{n \times D}$, and x_t^i is the data point at time step t. Time-series analysis aims to uncover patterns and relationships in X for tasks such as forecasting, classification, and anomaly detection. It plays a critical role in predicting complex phenomena, including weather patterns, energy consumption, traffic flow, and financial trends [25]. However, these applications present unique challenges, such as non-stationarity, high dimensionality, and long-range dependencies. Traditional statistical models such as ARMA and ARIMA capture trends and seasonality effectively but struggle with non-linear patterns [26]. Similarly, machine learning models, particularly RNNs and variants such as LSTMs and GRUs, are widely used [27], [28]. Nonetheless, these models process data sequentially, limiting their ability to capture long-range dependencies and making them susceptible to vanishing and exploding gradient issues [29], [30]. Furthermore, their sequential nature restricts the full utilization of modern hardware's parallel computation capabilities, such as GPUs and TPUs [31].

B. EMERGENCE OF TIME-SERIES LLMS

LLMs are advanced transformer-based models with billions of parameters, pre-trained on massive datasets [6]. These models excel in NLP tasks [7], [19], achieving state-of-the-art performance in comprehension, generation, and reasoning [20]. At the core of LLMs lies the Transformer architecture, the dominant framework for building LLMs [21].

Transformers leverage self-attention mechanisms to capture long-range dependencies and parallel relationships between tokens. This enables efficient processing of large-scale text data and scalable training [23]. The architecture has enabled the LLMs to push NLP performance to new heights, setting benchmarks across tasks and redefining the field [8].

Inspired by the success of LLM in NLP, researchers in the time-series field have started exploring the potential of LLMs for time-series analysis, leading to the development of time-series LLMs [11], [18]. These models leverage self-attention mechanisms to capture temporal dependencies and complex dynamics, effectively modelling input-output relationships [12], [32]. LLMs are well-suited for various time-series applications and can be trained from scratch on time-series data or adapted from pre-trained LLMs [33]. They have shown promise in tasks such as zero-shot time-series forecasting [34], classification [35], anomaly detection [36] and analysis [33]. Despite the lack of dedicated time-series datasets, pre-trained LLMs fine-tuned for time-series tasks are gaining popularity. Researchers address this limitation by leveraging LLMs trained on billions of tokens for tasks like classification, forecasting, and few-shot learning. These advancements are shaping the future of predictive analytics, offering innovative solutions to the challenges posed by evolving data complexities.

C. RELATED WORK

The field of time-series LLMs has seen increasing attention in recent years, with several reviews exploring various methodologies and applications. However, most existing studies lack a comprehensive systematic approach or focus on specific aspects of time-series analysis.

Su et al. conducted a review on transformer-based architectures for long-term time-series forecasting (LTSF), discussing key datasets, evaluation metrics, and the challenges inherent in forecasting tasks [17]. While their analysis provides valuable insights into the architectures, it does not fully address the role of tokenization strategies or explore the broader applications of time-series LLMs beyond forecasting. Similarly, Ye et al. reviewed time-series foundation models, proposing a framework for assessing model efficiency and explainability [11]. This study focused primarily on forecasting tasks and highlighted several key datasets and architectures but did not offer a systematic discussion of tokenization or expand the scope to include other time-series tasks such as anomaly detection or imputation.

The survey by [15] provides an overview of time-series forecasting models (TSFMs), including methodologies, pre-training techniques, and applications. However, it lacks a demographic analysis of the literature and fails to cover important tasks like classification and imputation, which are critical for time-series analysis. Zhang et al. [16] explored the challenges of adapting LLMs, trained on textual data, for time-series tasks, emphasizing the need for more efficient algorithms and better integration of domain knowledge. Although this work addresses important issues like domain adaptation, it does not offer a comprehensive taxonomy of time-series LLMs nor does it delve deeply into evaluation metrics or task-specific performance. In a similar vein, Jiang et al. [18] categorized LLM-based time-series methods into several areas such as direct queries, tokenization, prompt design, fine-tuning, and model integration. While this work provided insights into the applications of time-series LLMs across various domains, it fell short in offering a systematic synthesis of evaluation metrics or discussing the impact of tokenization strategies in detail.

Despite these contributions, existing reviews exhibit several limitations: lack a systematic framework, overlook key tokenization strategies, and predominantly focus on forecasting neglecting areas such as classification and imputation. Additionally, they fail to analyze demographic trends and research challenges like hallucinations and explainability. This study addresses these gaps through a comprehensive SLR, categorising and synthesizing contributions across tasks, tokenization strategies, datasets and metrics. It also offers an in-depth demographic analysis and identifies critical challenges in time-series LLMs.

D. THREAT TO VALIDITY

The limitations of this review have to be considered to have an overall analysis of the results gained from this SLR. Therefore, the key threats to the validity of this SLR are twofold, which are the potential incompleteness of the study search and biases in study selection. To address these concerns, this section provides a detailed discussion of each threat and the measures taken to mitigate them.

First, the study search was confined to five quality-controlled databases (Scopus, ACM, IEEE Xplore, Science Direct, and Springer) potentially excluding relevant studies from other venues. To mitigate this limitation, clear and well-defined inclusion and exclusion criteria were established, and supplementary sources were consulted to identify additional studies. Despite these efforts to enhance the completeness of the study search, the possibility of selection bias remains. This stems from excluding other databases such as Taylor & Francis, and Emerald Insight, which were not considered in this review.

Second, the study selection process relied on search strings defined by prior experience and focused on high-impact articles, which may have inadvertently excluded studies using alternative terminologies. To mitigate this, a comprehensive search strategy was developed using the PICO framework [39], complemented by a rigorous screening process guided by the PRISMA methodology [40], to ensure the inclusion of high-quality and relevant studies.

III. METHODOLOGY

This section outlines the methodology used in this SLR, as shown in Figure 1, following the guidelines proposed by [37]. The primary objective of this SLR is to identify, evaluate, interpret, and report on research related to

TABLE 1. Related work comparison.

Study	Year	Architecture	Tokenization	Task	Datasets	Evaluation Metric	Demographic Analysis	Challenges & Future Research	SLR
[17]	2023	✓	×	✓	✓	×	✓	×	×
[11]	2024	✓	×	✓	✓	×	✓	×	×
[15]	2024	✓	×	✓	×	×	×	×	×
[16]	2024	×	✓	✓	✓	×	×	✓	×
[18]	2024	×	✓	✓	×	×	×	✓	×
Our Work	2024	✓	✓	✓	✓	✓	✓	✓	✓

time-series LLMs. To achieve this, the methodology adopted is evidence-based and incorporates a rigorous study selection process to ensure transparency and reliability.

This paper addresses the growing interest among researchers in areas such as time-series architectures, tokenization approaches, time-series tasks, datasets, and evaluation metrics. Despite the increasing number of high-quality studies published in this research area, no SLR has specifically focused on time-series LLMs. To bridge this gap, this study undertakes a comprehensive SLR, guided by Kitchenham's well-established methodology. To ensure a successful and thorough review, the study was conducted systematically through several key phases: Planning, formulating Research Questions (RQs), conducting the Search Process, implementing the Study Selection Procedure, applying Inclusion and Exclusion Criteria, extracting Data, performing Quality Assessment (QA), and developing a Classification Scheme.

A. PLANNING

In this subsection, the scope and objectives of the study are clearly defined to provide a focused direction for the review. The process begins by identifying the key requirements of the study, ensuring that all critical aspects are addressed. Next, a detailed search strategy is designed, which includes developing precise search strings, formulating RQs, and selecting the relevant electronic databases for inclusion. To maintain relevance and currency, a publication window is established, targeting studies published between 2020 and 2024. This careful planning phase lays the groundwork for a structured and systematic research methodology, ensuring the review's rigour and comprehensiveness.

B. RESEARCH QUESTIONS (RQs)

This subsection presents the key RQs that shape this SLR. The primary aim is to provide a comprehensive review of time-series LLMs. These RQs define the scope of the review by addressing critical aspects that guide the evaluation process. To ensure relevance and depth, the RQs were meticulously crafted to align with the review's objectives, as detailed in Table 2.

C. SEARCH PROCESS

Formulating effective search strategies is essential for identifying key studies in an SLR [37]. To achieve this,

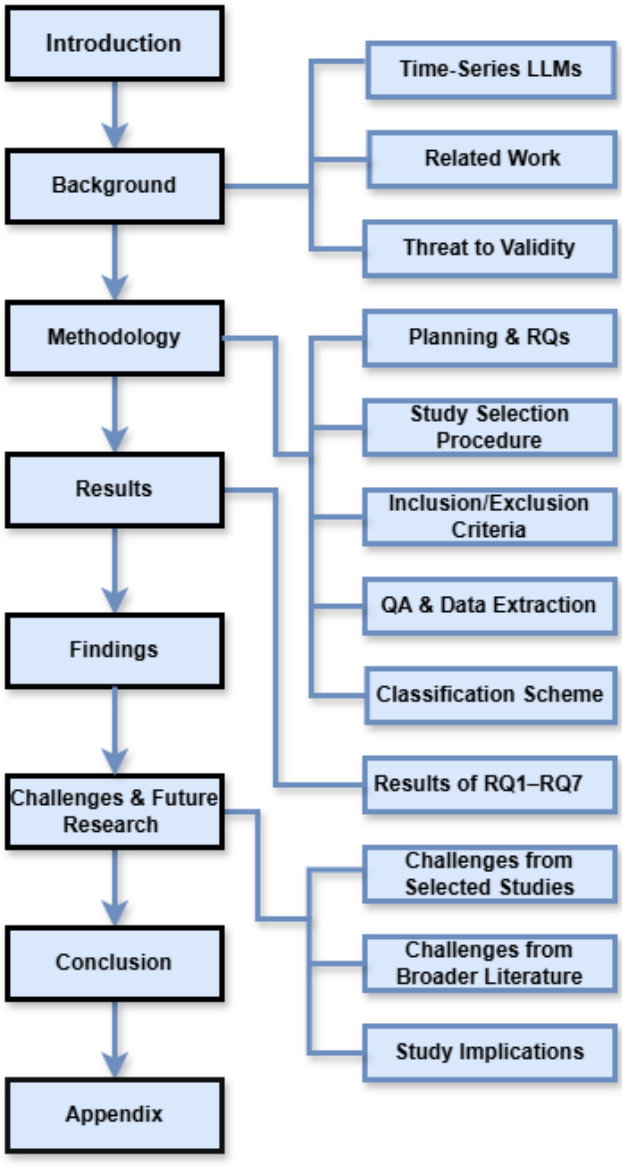

FIGURE 1. General overview.

[39] recommends the Population, Intervention, Comparison, and Outcome (PICO) framework, a widely adopted approach in many SLRs [40], [41]. Leveraging this framework, a standardized search string was created to ensure consistency across multiple databases. To maximize the retrieval of relevant articles on time-series LLMs, the following search

TABLE 2. Research questions and objectives.

S/N	Research Questions (RQs)	Objectives
1.	What are the demographic characteristics of the selected time-series LLM studies?	To understand the geographic distribution of the selected studies.
2.	What contributions have been proposed in time-series LLM research?	To identify, categorize, and analyze the contributions proposed in time-series LLM research.
3.	What transformer architectures have been used in existing time-series LLMs?	To analyze the existing transformer architectures used by the selected studies.
4.	How did the existing studies tokenize time-series data?	To investigate how time-series data is tokenized during pre-processing in time-series LLMs.
5.	What are the primary tasks addressed by the selected studies in time-series LLMs?	To understand the time-series LLMs' tasks in the selected studies.
6.	What are the datasets used in the selected studies?	To identify the datasets used to evaluate the time-series LLM research.
7.	What are the evaluation metrics used by the selected studies?	To understand metrics used to evaluate the time-series LLMs.

string was carefully designed: (Time-Series OR time-series OR Time Series OR time series) AND (Large Language Model OR LLM OR LLMs).

D. STUDY SELECTION PROCEDURE

To gather peer-reviewed literature, quality-indexed databases such as Scopus, ACM Digital Library, ScienceDirect, SpringerLink, and IEEE Xplore were utilized. These databases were selected for their extensive coverage of high-quality computer science research, ensuring the inclusion of significant contributions to the domain. A total of 754 documents were initially identified from the search process in these databases, and the results underwent a rigorous review to ensure their relevance to the study. To refine the selection, the PRISMA protocol was adopted as described in [105], enabling a systematic and transparent process. Duplicate entries identified across multiple databases were removed, leaving a final set of unique studies for further analysis, based on researchers' consensus.

From the IEEE Xplore database, 182 studies were initially retrieved. Titles and abstracts were screened, resulting in the exclusion of 148 unrelated documents and leaving 34 studies for a full-text review. After reviewing the full texts, 17 studies were included in the final analysis. ScienceDirect yielded 95 studies, which underwent title and abstract screening. This process excluded 76 irrelevant documents, leaving 19 for a full-text review. In the end, 12 studies met the inclusion criteria. In the ACM Digital Library, 96 studies were examined. After the screening phase, 57 documents were excluded, and 36 proceeded to a full-text review. Out of these, 16 studies were accepted for inclusion. SpringerLink produced 78 studies, of which 30 were excluded during the screening phase. From the remaining 38 studies subjected to a full-text review, 10 were included in the final analysis. Lastly, 182 studies were retrieved from other databases. After title and abstract screening, 70 documents were excluded, leaving 112 for a full-text review. From these, 18 studies were included in the final analysis. In total, this rigorous selection process resulted in the inclusion of 59 studies for the SLR. A detailed summary of the study selection process is provided in Figure 2.

TABLE 3. Inclusion and exclusion criteria.

S/N	Criteria	Inclusion	Exclusion
1.	Period	2020-2024	Before 2020
2.	Language	English	Non-English
3.	Type of source	Peer-reviewed journals, conferences, and workshops	Books and non-peer-reviewed articles
4.	Accessibility	Open access	Not open access
5.	Relevance	Relevant to the RQs	Not relevant to RQs

E. INCLUSION AND EXCLUSION CRITERIA

To ensure transparency and reproducibility, the inclusion and exclusion criteria outlined in Table 3 were carefully designed. These criteria were explicitly aligned with the RQs to prioritize open-access and high-quality studies. The inclusion criteria ensured that only studies focused on time-series LLMs, published between 2020 and 2024, peer-reviewed, and written in English were included. These studies were required to address at least two RQs to align with the study objectives and be accessible through established academic databases. Conversely, the exclusion criteria removed studies outside the specified period, unrelated to time-series LLMs, non-peer-reviewed, non-English, or inaccessible articles. These standards ensured a robust, high-quality dataset directly aligned with the research goals.

F. QUALITY ASSESSMENT (QA)

QA is crucial in any SLR, ensuring the identification of high-quality papers and filtering out irrelevant studies. It extends beyond initial inclusion and exclusion criteria, offering a more thorough screening process. This refined approach helps narrow the scope for data collection and analysis. In this study, QA focuses on assessing how well the selected studies address the defined RQs. To do so, a 1-4 scale questionnaire was developed, which provides the final QA score for each article (results are in the appendix).

- QA1: The paper highlights contributions to time-series LLMs and the tokenization approach, with response options: 'Yes (+1)', 'Partially (+0.5)', and 'No (+0)'.
- QA2: The paper outlines its transformer architecture and time-series task, with response options: 'Yes (+1)', 'Partially (+0.5)', and 'No (+0)'.

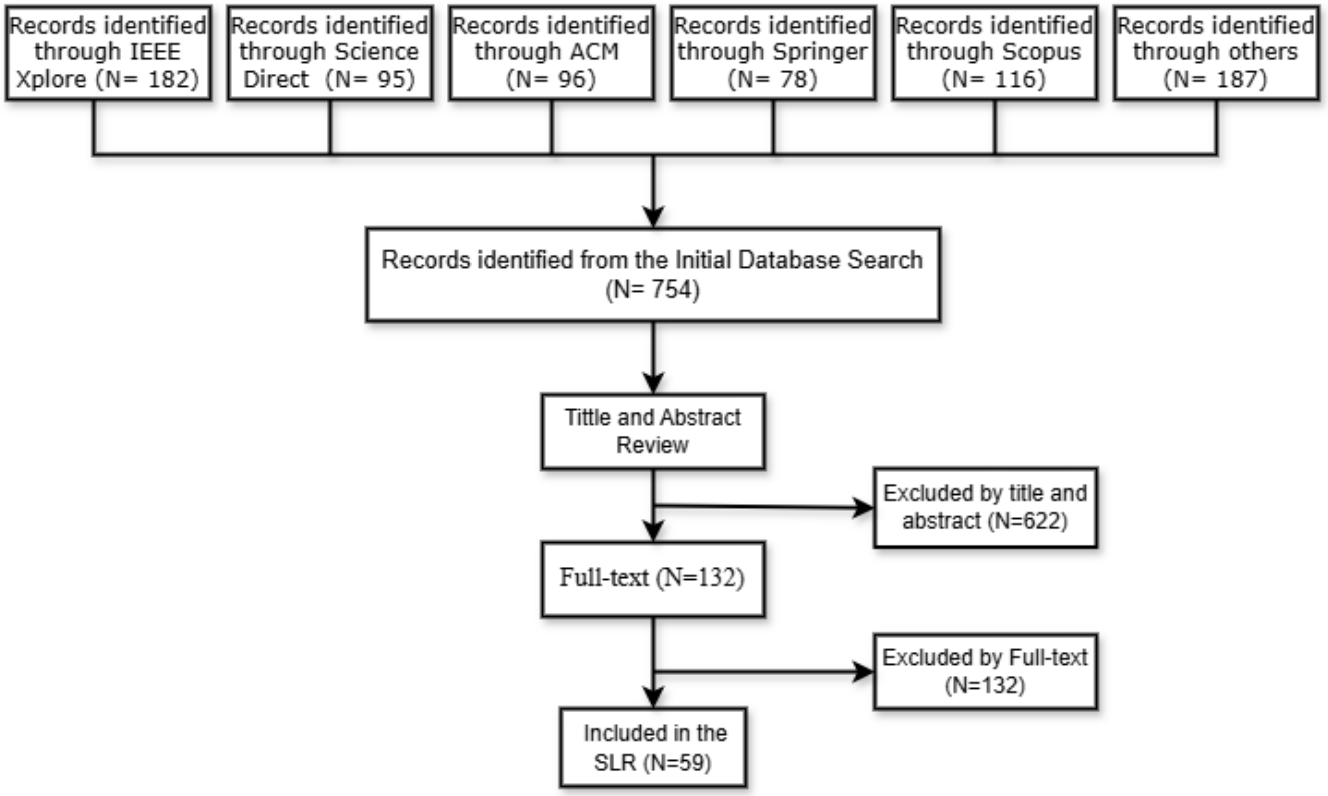

FIGURE 2. PRISMA protocol.

- QA3: The paper specifies datasets and metrics used, with response options: 'Yes (+1)', 'Partially (+0.5)', and 'No (+0)'.
- QA4: The paper outlines study limitations and future directions, with response options: 'Yes (+1)', 'Partially (+0.5)', and 'No (+0)'.

G. DATA EXTRACTION

In this section, the authors collaborated to systematically review the selected studies. The full text of each study was scrutinized by two researchers to extract relevant information based on the established RQs. A standardized data extraction form was used to capture the following key items:

- Title
- Publication year
- Publication venue
- Type of contribution
- Architecture
- Tokenization Approach
- Time-series Tasks
- Datasets
- Evaluation Metric
- Limitation

H. CLASSIFICATION SCHEME

This paper introduces a novel classification scheme for time-series LLMs, inspired by the work of [18], as presented in Figure 3. The scheme consists of six key dimensions: contribution, architecture, time-series tasks, tokenization approaches, datasets, and evaluation metrics.

For contributions, we identified six contributions from the selected studies:

1) Models: Novel time-series LLM models proposed in the literature.
2) Frameworks: Comprehensive time-series analysis frameworks incorporating LLMs.
3) Methods: Approaches and workflows for integrating LLMs into time-series tasks.
4) Approaches: Distinct algorithmic strategies or improvements within time-series LLMs.
5) Evaluations: Comparative analysis or benchmarking of time-series LLM performance.
6) Investigative study: Studies focused on exploring the application of LLMs in novel time-series contexts.

In the second classification, following the taxonomy proposed by [15], time-series LLMs are categorised into three main transformer architectures: encoder-only, encoder-decoder, and decoder-only. Encoder-only models provide full sentence access to attention layers at each stage [27]. Decoder-only models are auto-regressive, with attention limited to preceding words [91]. Encoder-decoder models function in a sequence-to-sequence manner, with encoder layers accessing the full sentence and decoder layers limited to prior words [90]. Additionally, Diffusion-based models

by [46] utilise stochastic processes to model data generation in time-series analysis.

In the third classification, based on [18], we categorized time-series LLM tokenization into seven approaches: String-based, Instance Normalization, Time-series Patching, Seasonal-Trend Decomposition, Cross-Modality Alignment, scaling, and quantization. However, in addressing RQ4, our study also identifies additional tokenization approaches from the selected studies that extend beyond this initial classification.

1) String-based: Tokenizing time-series data as strings, which are NLP-based approaches adopted for time-series.
2) Instance Normalization: Normalizing data at the instance level for a model to capture meaningful patterns
3) Time-Series Patching: Breaking the time series into patches for model input.
4) Seasonal-Trend Decomposition: Decomposing time series into seasonal and trend components before processing.
5) Cross-Modality Alignment: Aligning time-series data with other modalities.
6) Scaling and Quantization: Pre-processing data by scaling or quantizing for more efficient tokenization.
7) Digit Space: In digit-space tokenization, numbers are split into individual digits, enabling more efficient handling of both integers and floating-point values

In the fourth classification, as outlined by [11] and [16], we categorize the studies based on time-series tasks. The primary tasks identified in the selected studies include time-series forecasting, classification, imputation, anomaly detection and data generation. Each of these tasks addresses crucial challenges in managing and analyzing temporal data.

Time-series forecasting involves predicting future values based on historical data. Given a time series dataset $X = \{x_1, x_2, \cdots, x_n\} \in \mathbb{R}^{n \times D}$ (as defined in Section II-A), the objective is to forecast future values $(x_{n+1}, \ldots, x_{n+k})$, using a prediction function f parameterized by θ. This function leverages the historical data to model the temporal dependencies and project future outcomes.

In time-series classification, the goal is to assign a label c to a time series X, like detecting equipment states (normal or faulty). The predicted label C is determined by the classification function g and model parameters ϕ.

Time-series imputation estimates missing values from incomplete datasets. It leverages observed data and a mask matrix identifies observed and missing data points. In given time series X, a mask matrix $M \in \{0, 1\}^{n \times D}$ is used, where $M_{i,j} = 1$ indicates observable data and $M_{i,j} = 0$ indicates missing data. The imputation algorithm reconstructs the missing values to form $X^*_{i,j}$, represented as:

$$X^*_{i,j} = \begin{cases} \text{MV}, & \text{if } M_{i,j} = 0 \\ X_{i,j}, & \text{otherwise} \end{cases}$$

Time-series Anomaly Detection involves identifying outliers or irregular patterns within time-series that deviate from expected behavior. In a given time series X, the goal is to detect anomalies from the data points in the time-series. In time-series data generation, the model predicts the next point x_{t+1} from previous points $x_1, x_2, \ldots, x_t$ using: $x_{t+1} = f(x_1, x_2, \ldots, x_t) + \epsilon$ where f captures temporal patterns and ϵ adds variation.

In the fifth classification, we categorized the datasets used in the selected studies based on their application domains, following the approach of Su et al. [17]. The primary domains include Energy, Transport, Healthcare, Financial, Meteorology, Industry, Environmental, Multimodal, Anomaly Detection, Forecasting, and Classification.

The sixth classification categorizes metrics into five groups: regression metrics, classification metrics, specialized metrics, quality metrics, and domain-specific metrics. Metrics from the selected studies that do not fit into any of these categories are grouped under Others. Regression metrics (e.g., MSE, MAE, RMSE, MAPE) measure the error between predicted and actual values in time-series LLM forecasting tasks. Classification metrics (e.g., Accuracy, F1, Precision, Recall) assess the time-series LLM's performance in classification tasks. Specialized metrics (e.g., CRPS, MASE, FID) evaluate specific types of time-series analysis or generation tasks, such as probabilistic forecasting. Quality metrics measure computational efficiency and the quality of the output. Domain-specific metrics evaluate metrics tailored to specific application domains or types of time-series data.

IV. RESULTS AND DISCUSSION

This section presents the findings related to the RQs formulated in this study. A detailed breakdown of the results and the QA results is provided in Table 8 in the appendix.

A. RQ1: WHAT ARE THE DEMOGRAPHIC CHARACTERISTICS OF TIME-SERIES LLM STUDIES?

To address this RQ, we analyzed the 59 selected studies, focusing on publication trends and channels.

1) PUBLICATION TREND

Figure 4 shows the number of studies published between 2020 and 2024, highlighting the growing interest in time-series LLM research. In 2020, only two studies were published, followed by a notable decline in 2021, with no publications at all, making it the least active year. A slight increase occurred in 2022 with the release of just one study [43]. However, a significant surge followed in 2023 and 2024, with 12 studies in 2023 and an impressive 44 key studies in 2024. Despite the slow start, research in this field has gained significant momentum over the past two years. This increase can be attributed to breakthroughs in LLMs within NLP, which led researchers to apply these models to time-series tasks

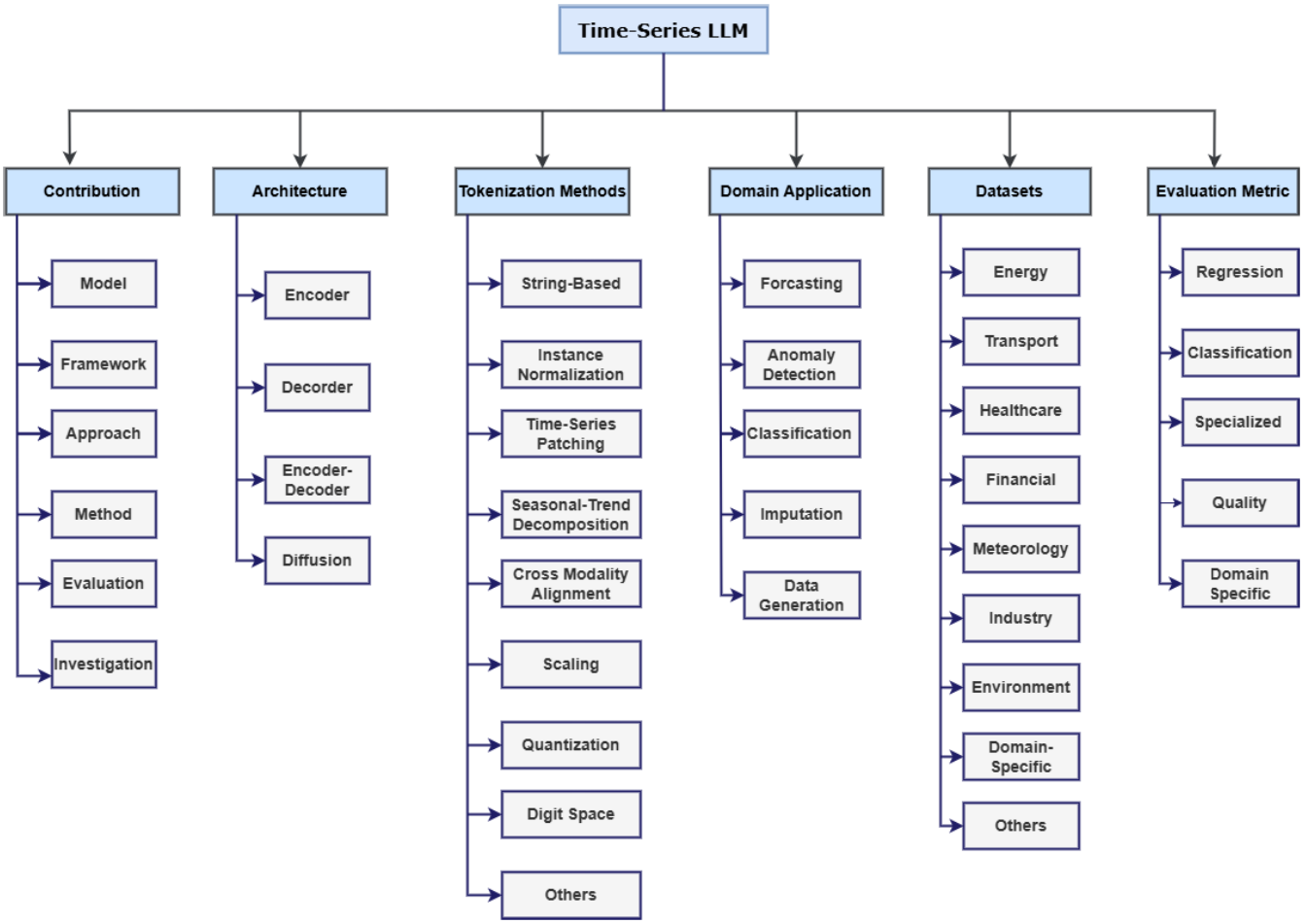

FIGURE 3. Classification Sscheme.

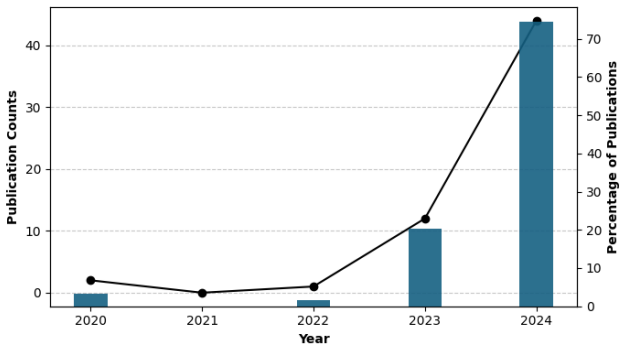

FIGURE 4. Publication trends.

2) PUBLICATION CHANNEL

From the 59 selected studies, 47 (79.66%) were published in traditional venues (conferences, journals, and workshops), while 12 (20.33%) were published as a preprint on arXiv [A5, A6, A10, A13, A22, A29, A43, A44, A55, A56, A57, A59] as presented in Table 4. The presence of preprints may be attributed to the fact that industry-based papers are often shared through arXiv rather than peer-reviewed venues.

Findings show that conference proceedings dominate the publication venues, with 33 papers accounting for 55.93% of the total studies. Journals follow, contributing 10 papers (16.94%), while workshops account for 4 papers, representing 6.77% of the total. Among the conferences, three key events have been particularly active in this field: the 41st International Conference on Machine Learning (ICML), the 37th Conference on Neural Information Processing Systems (NeurIPS), and the 2024 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ICML featured 7 papers, NeurIPS included 5, and ACM SIGKDD presented 4 studies, as shown in Table 5. This highlights the significance of these conferences as key venues for disseminating cutting-edge research in the time-series LLM field.

On the other hand, the limited journal publications (10 papers) in the research field raises concerns, as journals are generally regarded as more rigorous in terms of peer review compared to conferences [44]. This trend may be attributed to the relatively new nature of the research topic, where researchers prefer the faster dissemination of their work through conferences to maximize impact. In contrast, the journal publication process typically requires more time.

TABLE 4. List of studies by conference and journal.

S/N	Name	Studies	No
1.	41st International Conference on Machine Learning	A1, A3, A11, A20, A30, A42, A52	7
2.	37th Conference on Neural Information Processing Systems (NeuroIPS)	A12, A24, A25, A26, A53	5
3.	26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining	A4, A9, A15, A31	4
4.	37th AAAI Conference on Artificial Intelligence	A2, A45	2
5.	Annual Conference of the Association for Computational Linguistics (ACL)	A14, A23	2
6.	6th Artificial Intelligence and Cloud Computing Conference	A16	1
7.	34th International Conference on Tools with Artificial Intelligence (ICTAI)	A37	1
8.	12th International on Learning Representation (ICLR)	A8	1
9.	IEEE Conference on Computer Communication Workshop (INFORCOM WRSHP)	A34	1
10.	10th IEEE International Conference on Network Softwarization (NetSoft)	A35	1
11.	40th IEEE International Conference on Data Engineering Workshop (ICDEW)	A38	1
12.	25th IEEE International Conference on Mobile Data Management (MDM)	A40	1
13.	33rd International Joint Conference on Artificial Intelligence (IJCAI-24)	A7	1
14.	Causal and Object-Centric Representations for Robotics Workshop at CVPR 2024	A19	1
15.	Neural Information Processing Systems (NeuroIPS) 2023 Workshop R0-FoMo	A21	1
16.	International Journal of Simulation systems, science and technology	A27	1
17.	27th International Conference on Extending Database Technology (EDBT)	A28	1
18.	Workshop on Foundation Models in the Wild ICML 2024 FM-Wild Workshop	A32	1
19.	Journal of Communications and Information Networks	A36	1
20.	MDPI Journal of Energy	A48	1
21.	MDPI Journal of Sensor	A49	1
22.	MDPI Journal of Mathematics	A50	1
23.	Twelfth International Conference on Learning Representations	A51	1
24.	IEEE Transactions on Knowledge and Data Engineering Journal	A58	1
25.	Journal of Nature Portfolio	A17	1
26.	Transactions on Machine Learning Research (TMLR) Journal	A18	1
27.	ACM Transaction on Intelligent System and Technology Journal	A33	1
28.	IEEE 18th International Conference on Semantic Computing (ICSC) 2024	A39	1
29.	Findings of the Association for Computational Linguistics: NAACL 2024	A41	1
30.	Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing	A46	1
31.	Springer Journal of Data Mining and Knowledge Discovery	A47	1
32.	IEEE International Conference on Communications (ICC) 2024	A54	1
33.	arXiv	A5, A6, A10, A13, A22, A29, A43, A44, A55, A56, A57, A59	12

B. RQ2: WHAT CONTRIBUTIONS HAVE BEEN PROPOSED IN TIME-SERIES LLM RESEARCH?

In addressing this RQ, the study analyzed the contributions proposed by the selected papers. The findings revealed six main types of contributions: models, frameworks, methods, approaches, evaluations, and investigations (see Figure 5). Among these, models were the most frequently proposed, featured in 17 out of the 59 papers. Frameworks followed closely with 16 proposals, while approaches were identified in 12 studies. Evaluations appeared in 8 studies, methods in 5, and investigations in just 1 paper. In the following subsections, the respective studies are discussed in detail with respect to the proposed contributions.

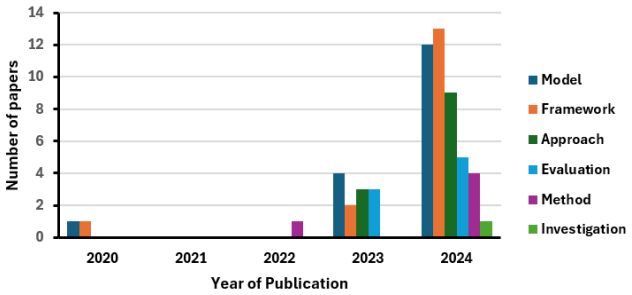

FIGURE 5. Contribution of the selected studies.

1) MODEL

We found that 17 out of the 59 selected studies proposed models in time-series LLMs. In a study by Wu et al., a model was developed based on a Generative Adversarial Network (GAN) architecture [45]. The model uses a sparse transformer to generate sparse attention maps, enhancing sequence-level prediction accuracy. Real-world experiments demonstrate that the model outperforms baseline models.

In another study by [46], the authors introduced a Time Diffusion Transformer (TimeDiT) model designed to tackle diverse real-world challenges in time-series analysis. The model utilized Transformer architecture to effectively capture temporal dependencies and incorporate diffusion processes to generate high-quality samples. Extensive experiments demonstrate the advancement of TimeDit in handling complex temporal patterns. Additionally, in a study by [47] a large pre-trained transformer model for analyzing financial time-series data named PLUTUS was developed. The model features an innovative invertible embedding module and multi-scale attention mechanisms that effectively capture intricate patterns while managing high noise levels. PLUTUS was pre-trained on a vast dataset of billions financial observations and its evaluation represents a breakthrough in financial time-series modeling. Furthermore, [48] proposed prompt-based time-series forecasting (PromptCast). This model transforms numerical inputs and outputs into natural language prompts, casting the forecasting task as a sentence-to-sentence generation problem. The evaluation

results of PromptCast showed promising results on the PISA dataset.

In another study, [49] proposed a pre-trained time-series model (LPTM) that employs an adaptive segmentation module, allowing it to represent diverse time-series data effectively across various domains. The model achieves effective performance in forecasting and classification while requiring less training time and data compared to traditional models. Similarly, Lag-Llama by Rasul et al. is a decoder-only transformer architecture that leverages lags as covariates [50]. Lag-Llama was pre-trained on a diverse corpus of time-series data from various domains. The results showcase strong zero-shot generalization capabilities and state-of-the-art results when fine-tuned on small, unseen datasets in comparison to the baseline models.

Additionally, [32] proposed a model-based lightweight TSMixer architecture named tiny time mixers (TTM). TTM consists of 1 million parameters and is exclusively trained on public time-series datasets, utilizing transfer learning for forecasting. The results comparison with state-of-the-art shows improved computing efficiency and reduced fine-tuning time. Das et al. proposed TimesFM, a foundational model for time-series forecasting that utilizes techniques from LLMs [51]. TimesFM employs a decoder-style attention architecture with input patching and a training regime tailored to varying context and horizon lengths specific to time-series data. The model learns temporal patterns, enabling accurate forecasting of unseen datasets. TimesFM achieves state-of-the-art zero-shot performance on public datasets, outperforming supervised models designed for each dataset. Eidele et al. proposed TSLANET, a lightweight adaptive network for time-series analysis that integrates convolution operations with adaptive spectral analysis [52]. TSLANET is a non-transformer-based architecture that excels in various analysis tasks. Comprehensive experiments in classification, anomaly detection, and forecasting demonstrated TSLANET' superior performance over traditional transformers.

The unified time-series model UNITS proposed by Gao et al. is a model capable of performing time-series forecasting, anomaly detection, imputation, and classification [53]. Results show UNITS excels in zero-shot, few-shot, and prompt learning tasks, outperforming task-specific models across various datasets. In a work by Pan et al., a model called Semantic Space-Informed Prompt Learning with LLM (S2IP-LLM) was proposed [54]. The proposed model aligned pre-trained semantic space with time-series embedding space to perform forecasting. When evaluated on seven M4 datasets, the S2IP-LLM demonstrated higher performance compared to existing models. Liu et al. proposed a spatial-temporal LLM (ST-LLM) for traffic prediction [55]. The model encodes time steps as tokens for each location and applies spatial-temporal embeddings to capture both spatial and temporal patterns effectively. Finding results indicate that the model performed robustly in both few-shot and zero-shot predictions.

2) FRAMEWORK

With regards to the framework, we observed that 16 (27.12%) studies proposed a framework. Fons et al. proposed a framework evaluating the capabilities of LLMs on time-series understanding [56]. The authors further introduced a taxonomy that categorized key time-series features. This taxonomy helps the authors in synthesizing existing time-series datasets. Hence, the general proficiency of LLMs in understanding time-series data was assessed. Based on the experiment conducted, the authors outlined the straight and limitations of existing LLMs in the time-series understanding by revealing the understood features by the models and features that falter.

In a work by Zhou et al., the authors proposed a generic time-series data synthesis approach for edge intelligence named GenG [57]. The proposed approach breaks the time-series data generation task into two, which are fine-tuning on LLM and using the transformer-based model to generate time-series data conditioning on the trained fine-tuned LLM model. The result shows significant improvement in efficiency in reasoning on generation tasks. The study by Zhou et. al proposes a Frozen Pretrained Transformer (FPT) Framework for time-series analysis [58]. FPT leverages a pre-trained LLM by freezing its self-attention and feedforward layers while fine-tuning the input embedding and normalization layers. FPT tackles the issue of limited time-series data for training by enabling the creation of effective pre-trained time-series LLMs. The evaluation results show that the FPT achieves state-of-the-art performance across various time-series tasks. Wang et al. proposed prompt-based domain Discrimination (POND), the first framework to use prompts for time series domain adaptation [59]. To facilitate learning, the proposed model compared a new instant-level prompt generator and fidelity loss mechanism to facilitate learning of meta-data information. The POND framework outperformed existing methods by up to 66% on the F1 score in extensive experiments on various datasets. In [60], a new framework that adapts LLMs for time-series representation learning (LLM4TS) was proposed. This framework redefines time-series forecasting as a self-supervised, multi-patch prediction task and introduces a patch-wise decoding mechanism. LLM4TS demonstrates effective performance across several downstream tasks, proving its capability to derive temporal representations with enhanced transferability. This represents a significant advancement in adapting LLMs for time-series.

A study by [61] proposed a time-series anomaly detection based on LLMs, named (LLMAD) framework. LLMAD accurately identifies anomalies while offering comprehensive interpretations to support decision-making. LLMAD employs few-shot anomaly detection by leveraging both positive and negative similar time-series segments for improved effectiveness. Experiments on three datasets reveal that LLMAD matches the detection performance of leading deep learning methods while providing superior interpretability. In another study, Du et al. proposed a multi-modal framework

PandemicLLM that transforms real-time disease spread forecasting into a text reasoning problem [62]. PandemicLLM was trained on public health policies, genomic surveillance, and epidemiological time-series data. The framework was tested in 50 states of the United States for 16 weeks and it effectively captured the impact of emerging variants and provided accurate predictions instantly.

Additionally, Wang et al. proposed a framework named K-link to address the issue of graph construction from MTS signals that introduces bias by the existing approaches [63]. The proposed framework adopts LLM to encode general knowledge to extract knowledge link graph that helps in capturing semantic knowledge of sensors, which helps reduce bias. Experimental results on multivariate time-series datasets demonstrate the proposed framework's effectiveness in improving performance on downstream tasks. Wu et. al. proposed a graph neural network (GNN) framework for multivariate time-series data [64]. The approach extracts relationships between variables that are uni-directional via a graph learning module. With this, external knowledge such as attributes or variables can be integrated. The authors also proposed a mix-hop propagation layer and dilated inception layer to capture spatial and temporal dependencies in time-series. The experimental result shows that the proposed framework outperforms the existing state-of-the-art.

Sayed et. al. proposed Gizaml, a framework designed for automated algorithm selection and hyper-parameter tuning for time-series forecasting. The demonstration results show some promise in time-series forecasting [65]. In a study by Cao et al., a framework named TEMPO was proposed [34]. TEMPO is a prompt-tuning-based generative transformer that is designed to learn time-series representations. The proposed framework uses two processes for time-series representation, decomposing the complex interaction between components and introducing prompts to facilitate adaptation. Experimental results demonstrate the framework's superior performance on zero-shot settings across multiple time-series datasets. Liu et al. proposed TimeCMA, a new LLM-based framework for time-series forecasting with cross-modality alignment [66]. Three modules were developed in the proposed framework, which are the dual-modality encoding module, cross-modality alignment module and time-series forecasting module. Experiments on the real datasets show the proposed framework achieved improved accuracy and efficiency.

In a study by Ansari et al., the authors proposed a framework called CHRONOS [67]. The framework tokenized time-series data into discrete bins by scaling and quantization of real values. With this, off-the-self language models can be trained on the language of time-series without changing the model architecture. The result shows the efficiency of the proposed framework on various time-series data in different domains, with improvement in zero-shot accuracy of unseen forecasting tasks. The study of Jia et al. proposed a multimodal framework for time-series forecasting, GPT4MTS, which leverages both numerical and textual information for improved predictions [68]. GPT4MTS uses textual summaries as soft prompts to guide the model in understanding context and influencing its forecasts. Findings show that combining textual information with numerical data enhances forecasting performance compared to traditional unimodal models. In the study of Chang et al., aLLM4TS framework was proposed, leveraging pre-trained LLMs to enhance the accuracy of time-series forecasting [69]. While LLMs trained on text data excel in few-shot learning, adapting them for time-series data poses challenges. LLM4TS addresses these issues by implementing a two-stage fine-tuning strategy: first, aligning the LLM with the unique characteristics of time-series data, followed by fine-tuning for specific forecasting tasks. Results demonstrate that LLM4TS outperforms existing methods, particularly in scenarios with limited data, indicating its potential for improving data-efficient forecasting in real-world applications.

3) APPROACH

From the selected studies, 12 proposed this approach, representing 20.34% of the total studies. The study in [70] proposed a novel approach LETS-C that leverages a language embedding model for time series data, paired with a simple classification head using CNNs and MLP, instead of fine-tuning LLMs. Experiments on benchmark datasets show that LETS-C outperforms current state-of-the-art models in classification accuracy while using just 14.5% of the trainable parameters, offering a more efficient solution.

Liu et al., propose an approach for time series anomaly detection using knowledge distillation and LLMs (AnomalyLLM) [36].The approach (comprising the student network and the teacher network) trains a student network to mimic the features of a teacher network, which is a pre-trained LLM adapted for time series analysis. AnomalyLLM detects anomalies when there is a significant discrepancy between the features of the teacher and student networks. Conclusively, the evaluation of the approach demonstrates good performance with high accuracy. Similarly, [35] proposed a novel approach named Instructime that reframes time-series classification as a learning-to-generate task. This approach incorporates task-specific instructions and raw time-series data as multimodal input, with label information encoded as text. The method comprises three key components: a time-series discretization module, an alignment projection layer, and autoregressive pre-training across domains. Experimental results demonstrate that Instructime outperforms baseline models in terms of classification accuracy and effectiveness.

Rehman et. al. proposed an approach coined adaptive contextual privacy preservation [71]. The approach analyses attributes in the dataset that are needed for specific application services. With this, sensitive attributes are identified to maintain the balance between privacy and service requirements. Experiments on power consumption and solar power generation datasets show that the proposed approach

achieves the set objective of the study. Liu et al. proposed an approach called Long-short-term Prompting (LSTPrompt) for zero-shot time-series forecasting tasks [72]. The approach divides time-series forecasting into short-term and long-term sub-tasks, allowing for tailored prompts for each sub-task. Evaluation of LSTPrompt indicates consistently strong performance across various forecasting scenarios. In another study by Liu et al., an approach named AutoTimes was proposed [73]. The aim is to repurpose LLMs as Autoregressive time-series forecasters. Furthermore, the proposed approach used time series as prompts, hence, extending the prediction context for an extended period. Experiment on multivariate time-series datasets shows the proposed approach achieved state-of-the-art performance.

4) METHOD

With regards to the method, five studies [A37, A27, A36, A47, A25] were proposed from the selected studies. Zhou et. al. proposed a meta In-Context Learning (M-ICL) method that utilizes LLMs to classify time-series electrical data without the need for annotated data when adapting to new tasks [74]. M-ICL leverages the strong in-context learning capabilities of LLMs to improve performance in electrical data classification. Experiments conducted on 13 real-world datasets show that M-ICL significantly enhances average accuracy across all datasets. Ceperic and Markovic proposed a method that improves LLM-based forecasting in time-series data [75]. The authors introduce an encoding strategy designed to align the quantitative characteristics of time-series data with the textual processing strengths of LLMs. This was achieved by using fast Brownian bridge-based aggregation (fABBA) algorithm. The experimental result shows the method on average improve time-series forecasting accuracy. In a study Tang and Zhang a new embedding method named MTSMAE [43]. The proposed method used a pre-trained approach based on masked autoencoder (MAE). The method demonstrates improved performance over supervised learning without pre-training. Evaluation using various multivariate time-series datasets shows that the proposed method outperforms existing approaches.

Graver et al. introduced a method called LLMTime, which enables efficient tokenization of time-series data and transforms discrete token distributions into adaptable continuous value densities [76]. The authors further show how LLM can handle missing data. The authors have shown that GPT-4 performed worse than GPT-3 because of how it tokenizes numbers. In a study by Liu et al., the authors proposed a new method called LLM-empowered channel prediction (LLM4CP) [77]. The proposed method predicts future downlink channel state information (CSI) sequences using past uplink CSI sequences as input. The proposed method used a pre-trained LLM model for the task, by fine-tuning the network while freezing most of the parameters of the pre-trained LLM. This is to have a better cross-modality knowledge transfer. Simulations conducted show that the proposed method achieves good prediction performance.

Chatzigeorgakidis et al. proposed a method approach for multivariate time series forecasting using LLMs, named MultiCast [78]. The method employs novel token multiplexing techniques to address handling one-dimensional data issues in time-series LLMs. Additionally, the paper introduces a quantization scheme based on the Symbolic Aggregate Approximation (SAX) to enhance LLM performance and reduce token usage for real-world applications. Evaluation of MultiCast against state-of-the-art methods demonstrates its effectiveness on real-world datasets. Su et. al. proposed a method for predicting resource needs in Virtual Network Function (VNF) environments using LLMs has been proposed to address the dynamic and non-linear challenge of traditional resource forecasting methods [79]. The method employs Llama2 to evaluate performance against statistical models on a public VNF dataset. The findings indicate that LLMs (Llama2) outperform statistical models in accuracy and efficiency.

5) EVALUATION

Eight (8) studies from the selected studies proposed evaluations for time-series LLMs. The study by Zhou et al. evaluates the performance of the LLMTIME model across multiple time series datasets [80]. Findings indicate that while LLMTIME demonstrates strong performance on certain datasets, it generally underperforms compared to traditional statistical models such as ARIMA, particularly when dealing with complex time series data. Ming Jin et al. conducted an evaluation study indicating that modern LLMs possess transformative potential for time-series analysis, particularly in supporting improved decision-making processes. [81]. The authors further explore various possibilities for LLMs in time-series advancements. In [82], an evaluation study was conducted to understand the extent to which destructive tools that are based on LLMs can correctly the source code needed to generate predictive models. The experimental results show that the composition of predictive models with complex architectures using LLMs is still far from improving the accuracy of predictive models generated by human data.

To ascertain if LLMs are effective on time-series data, Merrill et al. proposed an evaluation framework for time-series reasoning [83]. The authors evaluated 3 key aspects of whether LLMs achieved them. These aspects are etiological reasoning, question answering, and context-aided forecasting. The experimental result shows that LLM demonstrates limited time-series reasoning which shows the need for more research in the field. The authors in [84] conduct an evaluation study to investigate if LLMs are useful for time-series forecasting. The result of the study shows that pre-trained LLMs do not perform better than models trained from scratch despite their computational cost. The author further identified that the performance of LLMs in time series is similar. Ogawa et al. used explainable AI techniques to identify accident risks and suggest safer alternatives [85].

The authors combined knowledge graphs and large-scale language models for such a task. The results show some promise. The study by [86] establishes scaling laws for Time-series LLMs, showing that these models follow similar power-law scaling behaviour as LLMs in terms of parameter count, dataset size, and training compute. These findings provide insight into how increasing model size and computational resources can enhance Time-series LLM performance, much like the observed trends in LLMs. The study of [87] explores the potential of Time-series LLMs (TimeGPT) for load forecasting with limited historical data. The model's performance is compared to widely used machine learning and statistical models. Results show that TimeGPT surpasses these benchmarks, particularly in short-term predictions on real-world datasets with scarce training samples.

Finally, [88] explore the potential of time-series LLMs in financial time-series forecasting, using NASDAQ-100 stock prices as a test case. Experimental results in zero-shot and few-shot settings reveal that time-series LLMs can outperform traditional statistical and machine-learning models.

6) INVESTIGATIVE STUDY

In the investigative study, Mingyu Jin et al. compared the effectiveness of LLMs with traditional models for time-series forecasting [89]. The experimental results indicated that LLMs excel when applied to time-series data exhibiting clear patterns and trends associated with periodicity. Conversely, their performance diminishes when working with data that lacks periodicity. Additionally, the study found that incorporating external knowledge and employing natural language paraphrasing significantly enhances the predictive performance of LLMs.

C. RQ3: WHAT TRANSFORMER ARCHITECTURES HAVE BEEN USED IN EXISTING TIME-SERIES LLMs?

To address this RQ, we first analyze the core principles of the transformer architecture introduced by [90], which revolutionized neural machine translation by solving key challenges in sequence transduction.

The transformer was the first model to leverage the attention mechanism as its primary structure, eliminating the need for recurrent layers traditionally used in neural network architectures. The innovation of multi-headed self-attention allowed the transformer to efficiently capture long-range dependencies in sequences. Despite the wide range of transformer variations proposed in recent years for time-series analysis, our study classifies the selected studies according to the original architecture as defined by [90]. This decision underscores the foundational impact of the original transformer model, which inspired adaptations for time-series tasks such as forecasting, classification, and anomaly detection. By organizing the selected studies around this baseline, we provide a coherent framework for understanding the evolution of transformer-based models in the time-series domain.

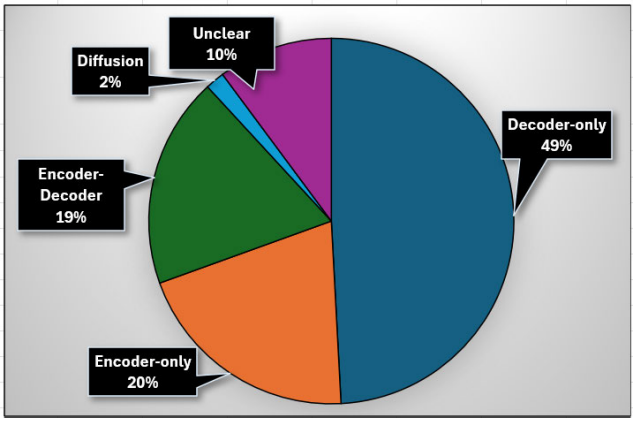

FIGURE 6. Transformer architecture used.

Transformer is an encoder-decoder architecture designed to convert a sequence of words from one language into a sequence in another language [91]. The architecture learns to encode the source sequence into a fixed-length representation, which is then decoded to generate the target sequence [92]. This process is carried out in an auto-regressive manner, meaning that information from later stages is propagated back to earlier stages during the generation of the translated sequence. This auto-regressive property is also applicable to time-series analysis. Transformers overcome the limitations of traditional auto-regressive architectures, which often struggle with long-term dependencies and experience information loss over time [27]. By leveraging self-attention mechanisms and positional encoding, Transformers offer a scalable and efficient solution for time-series modelling.

From the selected studies, we identified four (4) transformer architectures used by the selected studies in the field of study as presented in Figure 6. The architectures identified are decoder-only architecture with 29 studies, encoder-only architecture with 12 studies, encoder-decoder architecture with 11 studies and diffusion architecture with 1 study, respectively. The dominance of decoder-only architecture suggests a strong preference for Decoder architectures in time-series applications, while also indicating a diversity of approaches being explored in the research. Decoder architectures were predominantly used in forecasting tasks (A2, A5, A6, A8, A10, A13, A20, A21), with three studies focusing on anomaly detection (A7, A12, A26). A few studies (A12, A26, A32) applied decoder architectures for classification and imputation, further demonstrating their dominance in forecasting. We also observed that six (6) studies (A1, A3, A9, A15, A27, A11) were not clear on the architecture they utilized.

D. RQ4: HOW DID THE EXISTING STUDIES TOKENIZE TIME-SERIES DATA?

Tokenization plays a vital role in transforming raw time-series data into smaller, processable units essential for model training [48]. In analysing 59 selected studies,

we identified 22 tokenization techniques(see figure 7), showcasing the diversity of approaches in this domain. Among these, patching emerged as the most frequent method utilized in 18 studies. Patching appears either as a standalone technique [A3, A12, A20, A31, A36, A37] or in combination with other approaches [A6, A7, A8, A9, A10, A11, A19, A26, A30]. This technique segments time-series data into smaller patches, effectively capturing local patterns [18], [43], [77]. However, it can lose global temporal relationships when patches are overly small [111]. To address this, adaptive patching strategies that dynamically adjust segment sizes could mitigate such limitations [112].

The second employed method is instance normalization, used in 15 studies. It primarily appears in combination with other techniques [A7, A9, A10, A13, A27, A29, A44, A45, A46, A49, A52, A55, A58], with a few exceptions where it is used alone [A42, A58]. This technique excels in handling datasets with varying scales, significantly enhancing model performance [34]. The integration of patching and instance normalization [A6, A7, A8, A9, A10, A11] has been particularly effective for time-series data, as it simultaneously segments the data and normalizes it to ensure consistency across scales. Other notable approaches include scaling (10 studies), String-based (Byte Pair Encoding) (6 studies), and digit-space tokenization (4 studies), reflecting the influence of NLP methods on time-series tokenization. Although these NLP-inspired methods highlight the increasing convergence between NLP and time-series analysis, but often struggle to effectively capture the unique temporal dynamics of time-series data. Consequently, performance may be inferior to approaches specifically designed to address the complexities of time-series analysis [113]. To address these limitations, emerging approaches offer promising alternatives. Frequency-based encodings leverage spectral properties to efficiently capture periodic patterns [114], while segment-wise embeddings create contextualized representations of time-series segments, preserving temporal relationships akin to word embeddings in NLP [115]. Additionally, seasonal-trend decomposition further refines tokenization by tailoring it to distinct components of time-series data, such as seasonality and trends [79], [97]. These methods highlight the potential for task-specific innovations and the need for tailored solutions. Moreover, integrating advanced domain adaptation techniques into tokenization processes enhances their effectiveness. Pretraining on synthetic datasets (e.g., KernelSynth or TSMixup) mitigates data scarcity by exposing models to diverse temporal patterns, thereby enriching the robustness of tokenization [124]. Fine-tuning with task-specific datasets aligns tokenized representations with application needs, such as forecasting or anomaly detection [9]. Temporal and numerical embeddings, which capture seasonality, trends, and correlations, further enhance tokenization by enabling models to encode multi-dimensional relationships effectively [62].

This study also found that, among the selected studies, tokenization techniques such as quantization, sampling, lagged feature extraction, temporal embedding, and cross-modality alignment were each utilized in three studies. Additionally, Multi-Model Input Pipeline, Systematic Permutation of Input Sequence, Channel Independence, Prompt-Template, Indexing, Auxiliary Discriminator, fABBA, FBProphet, Segmentation, Standardization, and Scoring were each applied in one study, reflecting ongoing efforts to tailor tokenization to the diverse characteristics and challenges of time-series tasks. Notably, three studies (A34, A39, A57) did not specify the tokenization approaches used. Finally, our study observes that, despite the wide range of existing techniques, current tokenization methods may not fully address the diverse needs of time-series applications. This highlights the urgent need for more comprehensive tokenization strategies that can capture the complexities of time-series data and enhance the capabilities of time-series LLMs.

E. RQ5: WHAT ARE THE PRIMARY TASKS ADDRESSED BY THE SELECTED STUDIES IN TIME-SERIES LLMs?

In answering this RQ, the selected studies have been categorised into six main time-series tasks: forecasting/prediction, anomaly detection, classification, imputation, adaptation, and data generation. Figure 8 illustrates the distribution of these studies according to their primary tasks. Studies addressing multiple tasks are classified under all relevant categories.

Forecasting is the most prominent, with 47 studies, followed by classification with 9, anomaly detection with 8, and imputation and domain adaptation with 4 studies each. Data generation appears in just one study. This strong emphasis on forecasting highlights the primary focus on predictive modelling in time-series LLMs. However, the relatively limited focus on tasks such as imputation, domain adaptation, and data generation suggests significant opportunities for further exploration. Advancing research in these underexplored areas could lead to more versatile and robust time-series LLM models. Additionally, this study found that some studies [A3, A51, A4, A12, A26, A56, A42, A30, A22, A32] address multiple tasks, demonstrating the potential of time-series LLMs for broader applicability. Meanwhile, four studies [A11, A16, A14, A34] did not explicitly specify the tasks they addressed.

F. RQ6: WHAT DATASETS ARE USED IN THE SELECTED STUDIES

In this RQ, the analysis of 59 selected studies reveals the use of 263 datasets, 133 of which were unique. This dataset diversity reflects the broad range of data used to evaluate time-series LLMs' performance. Based on [17] classification, these datasets span multiple domains: Energy, Transport, Healthcare, Financial, Meteorology, Industry, Environmental, Multi-modal, and Domain-specific datasets.

Energy Domain datasets (such as Electricity, ETTh, ETTm, and ECL) were the most frequently used (31 studies), reflecting the importance of time-series LLMs for energy forecasting, particularly electricity demand

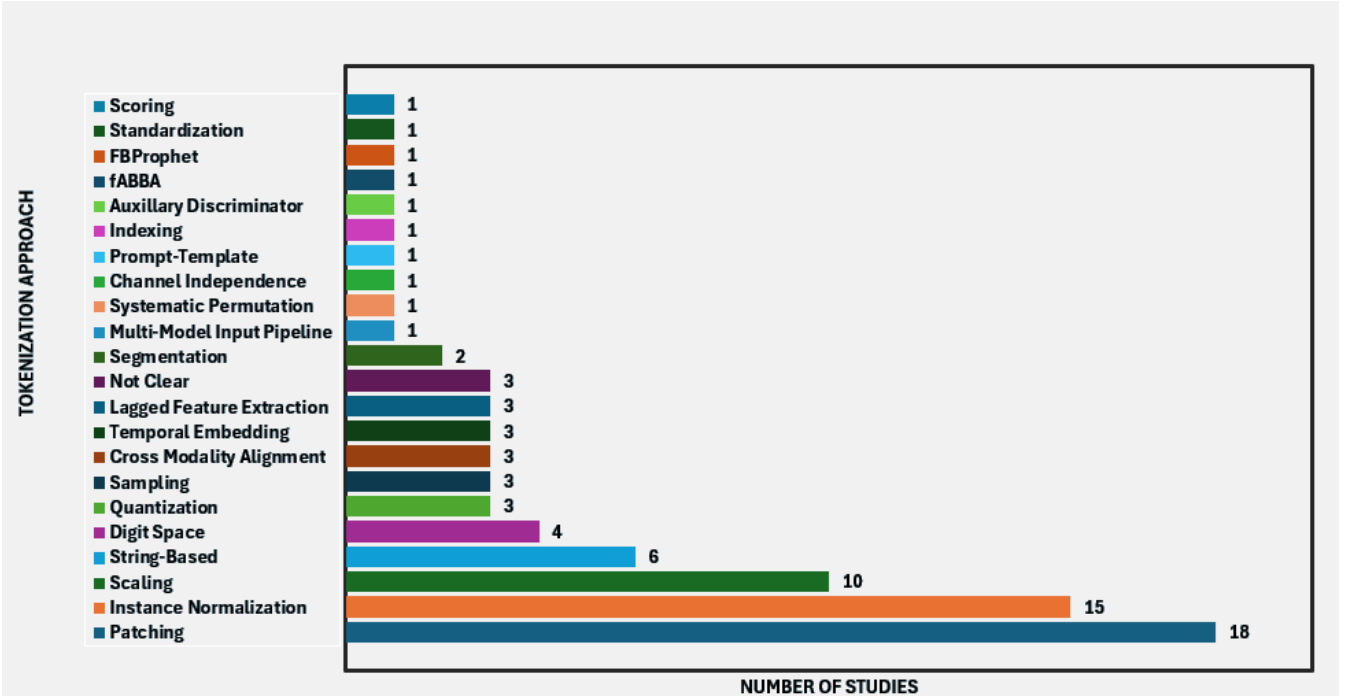

FIGURE 7. Tokenization approaches in the selected studies.

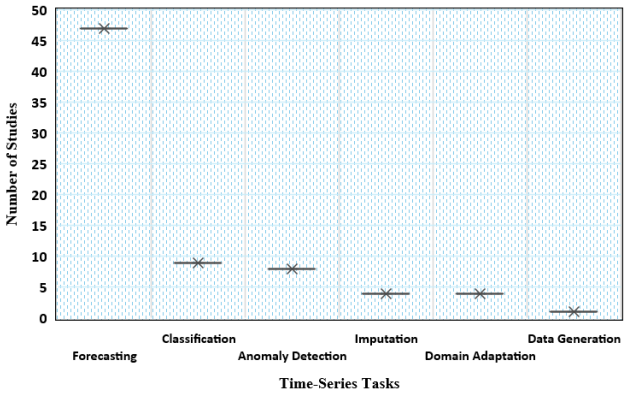

FIGURE 8. Time-series tasks in the selected studies.

prediction. Domain-specific datasets (e.g., Darts, Monash, Informer) were employed in 26 studies, showing a growing trend of using time-series-specific data for evaluation. Transport Datasets (Traffic, Taxi, PEMS, METRLA) and Meteorology Datasets (Weather, Climate, Temperature) ranked third, each appearing in 25 studies. The former highlights their relevance for traffic flow prediction and urban mobility analysis, while the latter emphasizes the increasing use of time-series LLMs for climate modelling and weather forecasting.

On the individual dataset level, ETTh and ETTm from the energy domain were the most frequently utilized, appearing in 15 studies (A1, A6, A10, A15, A26, A29, A30, A32, A37, A42, A43, A44, A45, A51, A53). Their prominence underscores their role as benchmark datasets for time-series forecasting, especially in energy-related applications. Weather datasets from the meteorology category were featured in 12 studies (A1, A6, A8, A10, A21, A26, A29, A30, A32, A37, A42, A44), primarily for weather forecasting and climate modelling tasks. Similarly, Traffic datasets from the transport category appeared in 11 studies (A1, A6, A8, A10, A15, A24, A26, A32, A42, A43, A44), demonstrating their importance in traffic prediction tasks. Electricity datasets were also significant, used in 10 studies (A1, A6, A10, A24, A26, A29, A32, A38, A42, A44) for electricity demand forecasting. Healthcare datasets, such as MIT-BIH, PSM, MSL, and SMAP, were notably utilized in studies (A3, A12, A19) focusing on health-related prediction tasks, reflecting the relevance of time-series LLMs in medical and biological time-series data analysis.

The reader needs to understand the widespread use of a combination of datasets (referred to as M4, comprising Traffic, Weather, ETTh, ETTm, and Electricity), which appeared in nearly 30% of the studies. This signifies the versatility of time-series LLMs in handling real-world challenges in energy, climate, and urban systems. In addition to real-world datasets, some studies utilized synthetic datasets to benchmark the performance of time-series LLMs, especially in cases where real-world data was scarce or difficult to access. The inclusion of domain-specific datasets, such as Monash for forecasting and UCR/UEA for multivariate time-series classification, further highlights the adaptability of time-series LLMs to diverse tasks beyond mere forecasting. These findings illustrate the broad applicability of time-series LLMs across diverse domains in energy, healthcare, and transportation and underscore their growing

effectiveness in tackling complex time-series tasks like forecasting, classification, and anomaly detection. These results affirm the importance of dataset diversity in evaluating the robustness and generalizability of time-series LLMs, driving advancements in both academic research and real-world applications. A detailed description of the datasets used in these studies is provided in Table 5.

G. RQ7: WHAT EVALUATION METRICS ARE USED BY THE SELECTED TIME-SERIES LLM STUDIES?

In response to this RQ, this study found that the 59 selected studies revealed a wide variety of evaluation metrics, with only one study [A11] lacking clarity regarding the utilized metrics (presented in Table 6). Significantly, 45 studies employed a combination of more than one metric, reflecting a diverse and comprehensive evaluation approach in the field of time-series LLMs. On the other hand, 14 studies utilized a single evaluation metric which are [A10, A13, A16, A20, A21, A22, A23, A24, A28, A39, A41, A42, A47, A48, A51].

Despite the wide range of metrics, the findings reveal that regression metrics are the most frequently used. Mean Squared Error (MSE) and Mean Absolute Error (MAE) appear in 29 studies each, followed by Root Mean Squared Error (RMSE) in 10 studies and Mean Absolute Percentage Error (MAPE) in 9 studies. MSE and MAE show a clear preference in energy and transport domains [54], [58], [60], [93], where they are valued for their effectiveness in quantifying prediction errors and evaluating forecasting models. In contrast, RMSE and MAPE are favoured for assessing forecast accuracy in the financial and activity recognition domains [64], [89]. It is worthy for the reader to note that the dominance of regression metrics (MSE, MAE, RMSE, MAPE) suggests a strong focus on prediction accuracy in time-series LLMs.

Secondly, following the regression metrics are classification metrics, with accuracy used in 14 studies, F1 score in 9 studies, then precision and recall in one study each. These metrics are frequently employed to evaluate the performance of predictive models, particularly in medical diagnosis within the healthcare domain [53]. Additionally, specialized metrics such as CRPS, Cosine Similarity, MASE, and FID were used in 4, 3, 2, and 1 studies, respectively. Notably, CRPS was primarily employed for probabilistic forecasts [50], while the remaining metrics were applied to specific types of time-series analysis or generation tasks. Quality metrics like execution time, perplexity, and similarity scores were used in one study each, highlighting a focus on computational efficiency and output quality. Domain-specific metrics, such as ROUGH, BER (Bit Error Rate), and Drunk-Dr, were also used in one study each, likely for specialized application domains or time-series data types. Other metrics, including RRSE, CORR, WMSE, RPS, Brier Score, WQL, Normalize Quantile Loss, SMAPE, R-Square, CVRMSE, IS, SE, WAPE, PEDS, Rtotal, Fidelity, Utility, Privacy, MSIS, MSPE, and Average Win, were each used in one study. In conclusion, the broad variety of metrics reflects the diverse applications and challenges present in the field of time-series LLMs.

V. FINDINGS AND DISCUSSION

This SLR aims to explore current knowledge in time-series LLM research. To achieve this, we selected 59 studies for analysis based on the methodology in Section III. These studies were thoroughly analyzed and synthesized to address the RQs detailed in Table 2.

Key findings from the SLR indicate a stable trend in publications over the past two years, as observed in the analysis of RQ1. Notably, there was a significant increase in research output in 2024, with 44 selected studies, compared to 12 studies in 2023, which accounts for 74.57% of the total studies. In contrast, there was only 1 study published in 2022 and two (2) in 2020, with no publications recorded in 2021. This sharp rise in research from 2023 to 2024 indicates that the field of time-series LLMs is rapidly emerging and gaining momentum. In response to RQ2, we identified six key contributions to time-series LLM research: Models, Frameworks, Approaches, Methods, Evaluations, and Investigative Studies. Also, our results highlighted a trend to argue that models, frameworks, and approaches were the most frequently proposed contributions by researchers in the domain, with 17, 16, and 12 studies, respectively.

To address RQ3, we categorized the selected studies based on the transformer architectures they employed: encoder-only, decoder-only, and encoder-decoder. Among these, the encoder-only architecture emerged as the most widely adopted, accounting for 49.15% of the studies. This preference is likely due to its efficiency in handling large-scale time-series data, making it particularly well-suited for tasks like forecasting and classification. Following this, the encoder-decoder architecture appeared in 22.33% of the studies, positioning itself as the second most used approach. Its strength lies in tackling complex sequence-to-sequence tasks, such as time-series forecasting and anomaly detection, where both input and output transformations are critical. Meanwhile, the decoder-only architecture was adopted in 18.64% of the studies. Although less prevalent, its effectiveness in autoregressive modelling tasks such as forecasting temporal dependencies demonstrates its value in specific scenarios.

Interestingly, we also observed a novel transformer variation: the diffusion model, utilized in 1.69% of the studies (e.g., [46]). This highlights the growing exploration of alternative architectures designed to address unique challenges in time-series analysis, such as incorporating external knowledge or managing uncertainty more effectively. While encoder-only architectures dominate due to their computational efficiency, encoder-decoder and diffusion-based models introduce scalability challenges that merit further attention. Encoder-decoder models, despite their strength in capturing long-term dependencies, suffer from quadratic complexity in attention computations, which makes them memory-intensive for long

TABLE 5. Dataset from the selected studies.

PS	Energy	Transport	Healthcare	Financial	Metrology	Activity Recognition	Industry	Domain Specific
A1	Electricity, ETTh, ETTm	Traffic	×	×	Weather	×	×	×
A2	×	×	×	×	GDELT	×	×	×
A3	×	×	MIT-BIH	×	×	HAR, UCI-HAR, WISDM, HHAR	×	UEA, UCR, SMD, MSL, SMAP, SWAT, PSM, UWAVE
A4	ISRIK-S3	×	×	×	×	UCR-HAR	×	C-MAPSS (FDoo2, FD004)
A5	×	Air Passenger, Traffic	HeartRate	Aus Beer, Monthly Milk	×	Sunspots	Gas Rate C02	Wine
A6	Electricity, ETTh, ETTm	Traffic	×	×	Weather	×	×	×
A7	×	×	×	×	×	×	×	UCR, SMD, SMAP, PSM, MSL, NIPS-TS-GELLO, NIPS-TS-SWAN
A8	ECL, ETTm, TETS	Traffic	×	EBITDA	Weather, GDELT	×	×	
A9	×	×	MED	×		HAR, HHAR, WISDM	×	SSL
A10	Electricity, ETTm, ETTh	Traffic	×	×	Weather	×	×	×
A11	×	×	×	×	×	×	×	×
A12	ECL, ETTm, ETTh, Solar	Traffic	Healthcare	Exchange Rate	Weather	×	×	SMD, MSL, SMAP, SWAT, PSM
A13	×	×	×	×	×	×	×	Synthetic Data
A14	×	×	×	×	×	×	×	Synthetic Data
A15	Solar, Electricity	Traffic, metr-la, Pems-bay	×	Exchange rate	×	×	×	×
A16	Power consumption, solar power generation data	×	×	×	×	×	×	×
A17	×	×	Public Health Data	×	Spatial Data, Genomic Surveillance	×	×	×
A18	Energy	×	Healthcare	Finance, Retail	Weather	×	×	Monash, M-Competition, Forecasting Repository
A19	×	×	FEG, ECG	×		HAR	FD, RWC	
A20	ETTm, ETTh	×	×	×		×	×	Darts, Monash, Informer
A21	ETT-M Square	Platform-Delay	×	×	Weather, Beijing-PM 2.5, PED-Counts	×	×	Request
A22	×	×	×	AB4, KPI	ABE	×	×	Yahoo
A23	×	×	×	×	×	×	×	Darts, Monash, Informer
A24	Electricity, Wind, Solar	Traffic	×	×	×	×	×	
A25		×	×	×	×	×	×	Darts, Monash, Informer
A26	Electricity, ETTh, ETTm	Traffic	IL1	×	Weather	×	×	UEA
A27	×	Air Passenger	Heart Rate	AUS Beer Monthly Milk	×	Sunspots	Gas Rate C02	Wine, Wooly
A28	×	×	×	Daily crude oil, Nasdaq, gasoline-price, water-price	Temperature	×	Month-c02, PPM/E	×
A29	ETTh, ETTm, ECL	×	IL1	×	Weather	×	×	×
A30	ETTh, ETTm	Traffic	IL1	×	Weather	×	×	×
A31	×	×	MED	×	×	HAR, HHAR, WISDM	×	SSL
A32	Electricity, ETTh, ETTm	Traffic	MED	×	Weather	×	×	×
A33	×	×	×	×	Temperature	×	×	×
A34	×	×	×	×	×	×	×	×
A35	×	×	×	×	×	×	×	UNF
A36	×	×	×	CSI	×	×	×	×
A37	ECL, ETTm, ETTh	×	×	×	Weather	×	×	×
A38	Electricity	×	×	×	Weather	×	Gas Rate	×
A39	×	×	×	×	Scenarios, Episode	Videos	×	×
A40	×	NYC Taxi, CHBike	×	×	×	×	×	×
A41	×	×	×	×	×	×	×	Wikipedia, WMT News
A42	ETTh, ETTm, ECL	Traffic, PEMS	×	×	Weather	×	×	
A43	ETTh, ECL, Solar	Traffic	×	×	Weather	×	×	

time-series sequences [119]. To address these challenges, several recent innovations have been introduced. First, Sparse Attention Mechanisms refine attention computation by focusing only on the most relevant tokens, reducing memory usage by assigning non-zero attention to a subset of tokens rather than computing scores for all token pairs,

TABLE 5. *(Continued.)* **Dataset from the selected studies.**

A44	ETTh, ETTm, Electricity	Traffic, Taxi	Covid death	Exchange-rate	Weather	×	×	NN5
A45	ETTh, ETTm, Electricity	Traffic	IL1	Exchange-rate	Weather	×	×	
A46	×	×	×	Stock	×	×	×	
A47	×	Skoda	PAMAP2, USC-HAD, Sleep, Epilepsy	×	×	×	×	WISDM2, WISDM
A48	Electricity	×	×	×	×	×	×	×
A49	×	Traffic	×	×	×	×	×	×
A50	×	×	M5, Stock	×	×	×	×	×
A51	ETTh, ETTm	PEMS-BAY, METRLA	Epidemic	Stock, Demand	×	×	×	UCI, UAE
A52	Electricity, ETTh, ETTm, Solar, Turkey power	Instanbul Traffic	×	Sales, Walmat	Weather	×	×	×
A53	ECL, ETTh1, ETTh2	Traffic	Illness	×	×	×	×	
A54	×	×	×	×	×	×	×	Darts, Monash, CEIE, Artificial Signal
A55	×	×	×	×	×	×	×	×
A56	×	×	×	×	×	×	×	×
A57	×	×	Health, PAMAP2	×	×	×	Industry	×
A58	ETTm, ETTh, ECL	Road Transport, PEMS	IL2, AF	Stock, Demand	×	×	×	×
A59	×	×	×	×	×	×	×	×

as is typical in traditional SoftMax-based attention [106]. However, as noted in [107], these mechanisms can sometimes dilute the focus on critical steps due to the SoftMax normalisation, potentially impacting performance. Second, Model Pruning has emerged as a practical solution by removing less significant layers or attention heads, thereby reducing the computational burden without significantly degrading model performance [108]. Third, Distributed Training by leveraging distributed architectures can accelerate training and allow the handling of larger sequences [109]. This approach is particularly relevant for encoder-decoder and diffusion-based models. Finally, Low-Rank Factorization decomposes attention weights into low-rank matrices, significantly reducing memory and computational demands and enabling broader scalability of the models [110].

In addressing RQ4, the tokenization approaches used in the selected studies were classified into seven main categories: String-based, Instance Normalization, Time Series Patching, Seasonal-Trend Decomposition, Cross-Modality Alignment, Scaling, digit space and Quantization as defined by [18].

From our analysis of 59 selected studies, Patching emerged as the most employed tokenization approach, utilized in 30.51% of the studies. This method segments time-series data into smaller patches to capture local patterns effectively. However, the risk of losing global temporal relationships when patch sizes are overly small underscores the need for more adaptive patching strategies. Instance Normalization ranked as the second most frequent approach, appearing in 27.12% of the studies. This technique is particularly adept at managing datasets with varying scales, often enhancing model performance. Scaling, the third most common approach, was observed in 16.95% of the studies, further emphasizing the importance of preprocessing techniques that normalize data for consistency across input sequences.

Interestingly, string-based techniques (traditional NLP approaches) were employed in 10 of the studies, reflecting a growing intersection between NLP and time-series analysis. Although NLP-inspired methods bring innovative perspectives, they often struggle to fully account for the unique temporal characteristics of time-series data, which can lead to suboptimal performance compared to techniques specifically designed for time-series tasks. The emerging approaches such as frequency-based encodings [114], segment-wise embeddings [115], and seasonal-trend decomposition [79] offer promising alternatives to the issue. Similarly, [123] recommends enhancing domain adaptation for numerical time-series through synthetic data pretraining, task-specific fine-tuning, and advanced embeddings. Our analysis also revealed a variety of emerging and innovative tokenization techniques tailored for time-series data, including Auxiliary Discriminator, Trainable Fully Connected Layers, Lagged Feature Extraction, Systematic Permutation of Input Sequences, Cross-Modality Alignment and fABBA Algorithm. These novel methods demonstrate the ongoing innovation in tokenization techniques tailored to the diverse requirements of time-series analysis.

For RQ5, we identified five primary categories of time-series LLMs tasks: forecasting, imputation, classification, anomaly detection, and data generation. Among these, the time-series forecasting task emerged as the predominant application, utilised in 79.66% of the selected studies. This indicates a significant emphasis on predictive modelling within the context of time-series LLMs. Following forecasting, the second most was time-series classification, explored

TABLE 6. Metrics and number of studies.

Metrics	No.	No. of Studies
MSE	29	A1, A2, A3, A5, A6, A8, A10, A12, A15, A17, A19, A26, A28, A29, A30, A32, A33, A35, A36, A37, A42, A43, A44, A45, A46, A51, A52, A53, A54, A56, A57, A59
MAE	29	A1, A2, A3, A4, A5, A6, A8, A12, A13, A15, A20, A23, A25, A26, A27, A29, A30, A32, A33, A35, A36, A37, A40, A43, A44, A45, A52, A54, A56, A57, A58
Accuracy	14	A3, A4, A7, A9, A12, A14, A17, A19, A25, A31, A39, A47, A48, A55
RMSE	10	A4, A15, A27, A32, A33, A35, A38, A40, A57, A58
F1	9	A3, A4, A7, A9, A14, A19, A22, A26, A31
MAPE	9	A5, A14, A15, A21, A27, A30, A33, A35, A40, A43, A57
CRPS	4	A21, A31, A52, A59
Cosine Similarity	3	A16, A36, A50
MASE	2	A18, A43
Precision	1	A7
Recall	1	A7
Nil	1	A11
RRSE	1	A15
CORR	1	A15
WMSE	1	A17
RPS	1	A17
Brier Score	1	A17
WQL	1	A18
Normalize Quantile Loss	1	A24
SMAPE	1	A30
R Square	1	A33
CVRMSE	1	A33
FID	1	A34
IS	1	A34
SE	1	A36
BER	1	A36
Execution Time	1	A38
WAPE	1	A40
Perplexity	1	A41
ONA	1	A43
ROUGH	1	A46
Drunk-Dr	1	A49
PEDS	1	A49
Rtotal	1	A49
Fidelity	1	A50
Utility	1	A50
Privacy	1	A50
MSIS	1	A52
MSPE	1	A53
Average Win	1	A55

in 15.25% of the selected studies. Anomaly detection was identified in 11.86% of the studies. These findings show the increasing focus on classification and anomaly detection as crucial tasks for time-series LLMs, alongside forecasting. A notable observation from our analysis is the diversity within the tasks. While forecasting is the main focus, researchers are exploring other tasks such as domain adaptation, imputation, analytical insights, and data generation. This suggests a broader range of potential applications for time-series LLMs in real-world scenarios. Furthermore, domain adaptation emphasizes the need to tailor models for specific contexts, enhancing their effectiveness in diverse environments. Data generation expands the use of time-series LLMs in synthetic data creation, invaluable for training models with limited data availability.

In RQ6, we identified that datasets from the Energy sector (including Electricity, ETTh, ETTm, ECL, and Solar) were the most frequently utilized, appearing in 31 studies. Following closely are domain-specific datasets (e.g., Darts, Monash, Informer) that were employed in 26 studies, while Transport Datasets (Traffic, Taxi, PEMS, METRLA) and Meteorology Datasets (Weather, Climate, Temperature) ranked third, each appearing in 25 studies. A significant observation is the combination of Traffic, Weather, ETTh, and ETTm datasets used in 30% of the selected studies, emphasizing the real-world applications of time-series LLMs and their importance in tackling sector-specific challenges. Additionally, we observed a positive trend toward dataset diversification, with many studies employing 4 to 5 datasets, and some using up to 13 datasets. This diversity strengthens model robustness and generalizability, enhancing the reliability of time-series LLMs across various domains.

As for RQ7, this study identified 28 evaluation metrics used across the 59 selected studies. Among these, MSE and MAE were the most frequently employed, with 29 studies utilizing them. The prominence of these metrics highlights a strong focus on prediction accuracy in time-series LLMs, particularly for tasks requiring precise forecasts. In contrast, other regression metrics such as RMSE and MAPE, which serve similar purposes as MSE and MAE, were less commonly used by the selected studies. For time-series classification tasks, F1-Score, Precision, Recall, and Accuracy were the most utilized metrics, reflecting a focus on the balance between correct and incorrect classifications. Additionally, the use of probabilistic metrics such as CRPS and Brier Score in some studies indicates an effort to incorporate uncertainty quantification into predictions, a growing area of interest in time-series analysis. Moreover, the inclusion of computational performance metrics, such as execution time, in certain studies [A38], demonstrates that efficiency and scalability are also key considerations in evaluating time-series LLMs.

VI. CHALLENGES AND FUTURE RESEARCH DIRECTIONS

The analysis of selected studies revealed key challenges and future directions in time-series LLMs. This section is structured into two subsections. The first discusses challenges and recommendations from the studies. The second explores additional challenges from broader literature.

A. CHALLENGES FROM THE SELECTED STUDIES

The analysis of selected studies revealed key challenges and opportunities in time-series LLMs. A notable trend is the adoption of NLP-inspired tokenization in 16.95% of studies, highlighting its crossover with time-series analysis. Although promising, these techniques often fail to address the unique temporal nature of time-series data, resulting in suboptimal performance compared to specialized methods [113]. To address this, future research should prioritize developing novel tokenization techniques tailored to the specific needs of time-series data [115]. This includes leveraging hybrid approaches that combine time-series-specific methods with NLP techniques and advanced domain adaptation strategies [79], [97]. The selected studies show a strong emphasis on time-series forecasting, with 79.66% of the studies utilizing this task. In contrast, time-series classification is represented in 15.25% of the selected studies. There is significantly less research on time-series data generation, anomaly detection, imputation, and domain adaptation tasks. Therefore, future research on time-series LLMs should prioritize data generation, anomaly detection, and imputation to provide a more comprehensive understanding of their capabilities. Datasets from Energy (31 studies), domain-specific (26 studies), transport (25 studies), and meteorology (25 studies) dominate time-series LLM studies, potentially limiting model generalizability to fields like healthcare, finance (17 studies), Activity recognition (9 studies), manufacturing (5 studies). Future research should explore broader datasets to enhance model applicability across diverse domains. MSE and MAE (used in 49% of the selected studies each) emerged as the most commonly employed regression metrics, followed by RMSE (16.94%), MAPE (15.25%) and MASE(3.38%), which were utilized less frequently. However, metrics such as SMAPE were absent from any selected studies. Future research should explore the adoption of these underutilized metrics to diversify evaluation methods in time-series LLM research.

B. CHALLENGES FROM THE BROADER LITERATURE

This sub-section outlines key challenges identified from the broader literature, providing insights into current limitations and highlighting potential avenues for future research:

1) **Hallucinations:** Hallucination is a significant issue in LLMs, raising critical concerns when these models are applied to sensitive domains like time-series analysis, where accuracy is paramount. Hallucination refers to the tendency of LLMs to generate incorrect or fabricated content, often leading to misleading outcomes in critical applications [104]. The existence of hallucinations could undermine the performance of LLMs and pose a significant barrier to LLMs adoption in real-world applications by eroding user trust [121]. Hallucinations often stem from challenges such as unclear prompts, biased training data, or inaccurate LLM parameters, highlighting the complexity of tackling this issue [125]. To address these issues in traditional LLMs, researchers have explored solutions such as prompt engineering, dataset refinement, and model adjustments, with varying degrees of success [104]. Despite these efforts, no study has yet focused on the critical task of detecting and mitigating hallucinations in time-series LLMs. This gap presents an exciting opportunity for researchers to pioneer innovative solutions for detecting and mitigating hallucinations in time-series LLMs.
2) **Long-term dependency:** The rising popularity of time-series LLMs is attributed to the ability to capture long-range dependencies and interactions within data, resulting in impressive prediction performance [14]. However, difficulties arise in understanding complex patterns, particularly when handling large volumes of univariate or multivariate data across various domains [11]. To address these challenges, this study recommends enhancing the modelling of dependencies in time-series LLMs through adaptive segmentation tokenization that dynamically adjusts segment sizes to preserve critical temporal patterns [120]. This ensures tokenization aligns with varying temporal scales, allowing LLMs to capture dependencies across different levels of granularity. Additionally, adopting data augmentation methods can increase data diversity, enabling models to generalise more effectively across temporal patterns. At the same time, multi-scale temporal aggregation embeds hierarchical representations of time-series data. Finally, autoregressive prediction strategies leverage the sequential nature of LLMs to handle long-term dependencies and reduce error accumulation in extended forecasts.
3) **Tokenization:** Tokenization is a crucial preprocessing step in time-series LLMs, serving as the foundation for effectively capturing temporal dynamics and tackling the challenges of non-stationarity inherent in time-series data [52]. The most commonly used tokenization approaches in time-series LLMs (such as normalization, patching, scaling, and quantization) are designed to capture local patterns but often risk overlooking global temporal relationships [34], [36], [96]. Similarly, string-based methods adopted from NLP, often fall short in encoding complex, multi-scale temporal dependencies [76]. To address these limitations, this study recommends that future researchers integrate emerging approaches [114] and incorporate advanced domain adaptation techniques into the tokenization process [124], thereby better capturing temporal dynamics and advancing applications across diverse time-series tasks.
4) **Explainability:** Despite the successes of LLMs in various applications, these models remain opaque in their decision-making processes, often producing untraceable outputs that raise significant concerns about explainability. In time-series LLMs,

explainability is key to understanding model behaviour and predictions, which is a critical factor for fostering trust, especially in high-stakes fields like healthcare and finance. Although time-series LLMs leverage advanced techniques like tokenization, prompts, embeddings, and fine-tuning to enhance performance, these models are often regarded as black boxes. This lack of interpretability limits the ability to explain the rationale behind predictions, which is essential for decision-making in high-stakes applications. Attention mechanisms excel at capturing time-based patterns, making them powerful tools for analyzing temporal dependencies. However, the lack of interpretability in time-series contexts hinders understanding and application. To enhance explainability, future research can integrate interpretable attention mechanisms, such as sparse attention [110] and self-attention visualization [117], enabling users to identify key inputs during predictions. This will improve decision-making, allow LLMs to prioritize task-specific information, and enhance reasoning capabilities [116]. Additionally, post-hoc methods like SHapley Additive Explanations and Integrated Gradients offer temporal attributions by identifying features or time points crucial to the model's output, aligning focus with clinically or financially significant patterns [118].

5) **Scalability:** Despite the effectiveness of the transformer architecture in capturing long-term dependencies, encoder-decoder models face scalability challenges, particularly for long sequences. The attention mechanism, though powerful, is memory-intensive, requiring the storage of attention weights for all input token pairs, leading to scalability issues [119]. Moreover, in some time-series tasks such as forecasting, only a few historical steps are typically relevant, making it inefficient to allocate attention to irrelevant steps. Sparse Transformers provide a partial solution by concentrating attention on the most relevant steps, reducing memory usage [122]. However, their reliance on SoftMax-based normalization can still dilute the focus on critical steps, potentially affecting model performance. These challenges highlight the need for further innovations in attention mechanisms and token relevance optimization to enhance the scalability and efficiency of transformer architectures for time-series tasks.
6) **Bias and Safety:** Generally, LLMs exhibit bias during model training and show a strong preference for specific tasks. There have been no research efforts aimed at addressing bias in time-series LLMs, despite the numerous challenges and future opportunities. This study recommends novel debiasing techniques to mitigate biases in time-series analysis and improve the performance of time-series LLMs.
7) **Multi-Modality:** Time-series data can be integrated with data from other sources. Time-series LLM studies often treat time-series data as text sequence inputs or align time-series inputs with LLM textual embeddings [33]. Very few approaches have utilised multimodal inputs containing both time-series and textual information. This study's findings show the use of the Traffic, Weather, ETTh, and ETTm datasets in 30% of the reviewed studies. This trend underscores a strong focus on real-world, practical applications of time-series models, highlighting the need to incorporate multimodal inputs via LLMs, align different modalities in the embedding space, and interpret the output accordingly. To address these issues, new multimodal time-series LLMs are needed by the researchers. Cross Modality Alignment techniques may help tackle these challenges.

C. IMPLICATION OF THE STUDY

This study offers valuable insights into advancing the field of time-series LLMs. Despite prior contributions, critical gaps persist in areas like demographic analysis and discussions on innovations and challenges, including long-term dependency, hallucinations, scalability, explainability, and temporal tokenization. The paper provides a novel classification framework that highlights advances in tokenization, datasets, and evaluation metrics, forming a foundation for more efficient and interpretable models. The findings highlight the transformative potential of these models in overcoming challenges such as long-range dependencies, high dimensionality, and non-stationarity. The study advocates for diversifying tasks beyond forecasting, such as classification, anomaly detection, and imputation. Similarly, it recommends expanding datasets beyond energy and transport domains to enhance generalizability. By filling these gaps, time-series LLMs can become transformative tools in fields like energy management, climate modelling, and financial risk analysis, enabling precise, large-scale decision-making.

VII. CONCLUSION

This SLR analyses research contributions in the time-series LLMs field from 2020 to 2024. Time-series LLMs have become essential in time-series analysis, excelling at capturing complex patterns and long-range dependencies in time-series tasks such as forecasting, anomaly detection, and classification. With rapid advancements in this area, a thorough review of the state-of-the-art research is crucial for understanding the current scope and identifying future directions. To bridge existing gaps, this review adopts an evidence-based methodology, starting with 754 articles retrieved from the initial search, which were subjected to strict inclusion criteria and QA. This rigorous process led to the selection of 59 high-quality studies, systematically categorized by contributions, architectures, tokenization techniques, tasks, datasets, and evaluation metrics.

The results indicate a growing focus on time-series LLM research over the past two years, with 55.93% of selected studies presented at conferences, followed by journals

(16.94%) and workshops (6.77%). The analysis and synthesis revealed six key contributions, with model proposals (17 studies) and framework proposals (16 studies) being the most common. Moreover, from the transformer architecture perspective, decoder-only models were the most prevalent, featured in 29 studies, followed by encoder-only (13 studies), encoder-decoder (11 studies), and diffusion-based architectures (1 study). Additionally, patching was the most frequently used tokenization approach (30.51%), followed by instance normalization (25.42%), and scaling (16.95%). As for datasets, energy-related datasets such as electricity, ETTh, ETTm, and ECL were the most frequently utilized (31 studies), then domain-specific (26 studies), followed by those from transport and meteorology (25 studies each). These findings highlight a strong reliance on domain-specific datasets for time-series tasks. Regarding evaluation metrics, regression metrics (MAE and MSE) were predominantly used, followed by classification metrics, with fewer studies specialized and quality metrics. Findings further revealed gaps, including the need to address hallucinations, encode complex multi-scale and long-range temporal dependencies, and develop scalable and explainable time-series LLMs, as emphasized by researchers in the field. Future research should focus on adaptive tokenization methods, enhancing explainability to improve trust and usability, and diversifying datasets to include domains such as healthcare and finance. Additionally, expanding evaluation metrics to address hallucination challenges is essential for advancing time-series LLM performance.

In general, given the strong interest in time-series LLM research and the recent consistency in publications, more concrete solutions to issues such as hallucination, tokenization, and long-range dependencies are expected in the coming years. Research challenges and potential directions for future work are further discussed in detail in Section VII. Therefore, researchers must focus on these areas to effectively address the underlying challenges and propose actionable solutions.

TABLE 7. Glossary of key terms.

Term	Definition
LLM	Artificial intelligence models trained on vast amounts of text data to understand and generate human-like language.
Time-series LLMs	LLMs designed to analyse time-series data by capturing complex patterns and dependencies over time.
Pre-trained model	An LLM that has already been trained on a large dataset and is ready for use in solving similar tasks without requiring training from scratch.
Self-attention Mechanism	A technique used in transformer-based models where each data point is compared to all other points in the sequence to determine the most relevant ones, regardless of their distance in the sequence.
Long-range Dependencies	The relationship between data points in time-series data that are spaced far apart but still influence each other.
Tokenization	The process of converting time-series data into tokens that the model can process.
Patching	A tokenization technique where the time-series data is divided into patches to improve efficiency in processing.
Hallucination	Generation of incorrect or fabricated information that doesn't align with the ground truth.
Transformer Architecture	An LLM architecture that primarily relies on self-attention mechanisms to process and analyse input sequences in parallel.

LIST OF ABBREVIATIONS

ABE Accuracy-Based Evaluation.
ARIMA Autoregressive Integrated Moving Average.
BPE Byte Pair Encoding.
CEIE Critical Energy Infrastructure Evaluation.
CORR Correlation Coefficient.
CRPS Continuous Ranked Probability Score.
CVRMSE Coefficient of Variation of Root Mean Squared Error.
Darts Demand-Adjusted Residual Time-Series.
Drunk-Dr Drunkard's Walk Detection Rate.
ECL Electricity Consumption Load.
ETTh Electricity Transformer Hourly.
ETTm Electricity Transformer Minute.
FID Fréchet Inception Distance.
FPT Frozen Pretrained Transformer.
GAN Generative Adversarial Network.
GDELT Global Database of Events, Language, and Tone.
GPU Graphics Processing Unit.
GRU Gated Recurrent Unit.
HAR Human Activity Recognition.
HHAR Hierarchical Human Activity Recognition.
IS Inception Score.
KPI Key Performance Indicator.
LlaMA Large Language Model Meta AI.
LSTM Long Short-Term Memory.
MAE Mean Absolute Error.
MAPE Mean Absolute Percentage Error.
MASE Mean Absolute Scaled Error.
MED Mean Energy Deviation.
METRLA Metropolitan Transit Authority of Los Angeles.
MSE Mean Squared Error.
MSIS Mean Scaled Interval Score.
MSL Multi-Source Learning.
MSPE Mean Squared Prediction Error.
NN5 Neural Network 5.
NLP Natural Language Processing.

TABLE 8. Appendix.

ID	Ref	Year	Architecture	Tokenization Process	QA1	QA2	QA3	QA4	Total Score
A1	[54]	2024	Not Clear	Cross Modality Alignment Module	1	0.5	1	1	3.5
A2	[68]	2024	Decoder	Multi-Model Input Pipeline	1	1	1	1	4
A3	[52]	2024	Not Clear	Patching	1	0.5	1	1	3.5
A4	[63]	2024	Encoder	Graph Module alignment for Time-series	1	1	1	1	4
A5	[89]	2024	Decoder	Scaling, Systematic Permutation	1	1	1	1	4
A6	[93]	2024	Decoder	Patching, Channel Independence	1	1	1	1	4
A7	[36]	2024	Decoder	Patching, Instance Normalization	1	1	1	1	4
A8	[34]	2023	Decoder	Patching, Encoding	1	1	1	1	4
A9	[59]	2023	Not Clear	Patching, Instance Normalization	1	0.5	1	1	3.5
A10	[32]	2024	Decoder	Patching, Instance Normalization	1	1	1	1	4
A11	[81]	2024	Not Clear	Patching, Tokenizer	1	0	0.5	1	3.5
A12	[53]	2024	Decoder	Patching	1	1	1	1	4
A13	[83]	2024	Decoder	Instance Normalization, Scaling	1	1	1	1	4
A14	[56]	2024	Decoder	Digit space, BPE	1	0.5	1	1	3.5
A15	[64]	2020	Not Clear	Graph Module alignment for Time-series	1	1	1	1	4
A16	[71]	2023	Decoder	Prompt-Template	1	0.5	1	1	3.5
A17	[62]	2024	Encoder	Temporal Embedding	1	1	1	1	4
A18	[67]	2024	Encoder-Decoder	Scaling, Quantization	1	1	1	1	4
A19	[35]	2024	Encoder-Decoder	Patching, Encoding	1	1	1	1	4
A20	[51]	2023	Decoder	Patching	1	1	1	1	4
A21	[50]	2024	Decoder	Lagged Feature Extraction	1	1	1	1	4
A22	[61]	2024	Encoder-Decoder	Rescaling, Indexing	1	1	1	1	4
A23	[71]	2024	Decoder	Time-series data Decomposition	1	1	1	1	4
A24	[45][	2020	Encoder-Decoder	Auxiliary Discriminator	1	1	1	1	4
A25	[76]	2024	Decoder	BPE, Add Space, Rescaling, Sampling	1	1	1	1	4
A26	[58]	2023	Decoder	Patching, Instance Normalization	1	1	1	1	4
A27	[75]	2024	Not Clear	fABBA	1	0.5	1	1	3.5
A28	[65]	2024	Decoder	Sampling, FBProphet, Lagged Feature Extraction	1	1	1	1	4
A29	[66]	2024	Encoder	Instance Normalization, Cross Modality Alignment, Token Embedding	1	1	1	0.5	3.5
A30	[60]	2024	Encoder-Decoder	Patching, Sampling	1	1	1	1	4
A31	[94]	2024	Encoder	Patching	1	1	1	1	4
A32	[46]	2024	Diffusion	Segmentation, Standardization	1	1	1	1	4
A33	[82]	2024	Decoder	Lagged Feature Extraction	1	1	1	1	4
A34	[80]	2024	Decoder	Not Clear	0.5	0.5	0.5	1	2.5
A35	[79]	2024	Decoder	Time-series data Decomposition	1	1	1	1	4
A36	[77]	2024	Encoder	Patching	1	1	1	1	4
A37	[43]	2022	Encoder-Decoder	Patching	1	1	1	1	4
A38	[78]	2024	Decoder	Quantization	1	1	1	0.5	3.5
A39	[85]	2024	Encoder	Not Clear	0.5	1	1	1	3.5
A40	[55]	2024	Decoder	Spatial Temporal Embedding	1	1	1	1	4
A41	[95]	2024	Decoder	BPE	1	1	1	1	4
A42	[96]	2024	Encoder-Decoder	Instance Normalization	1	1	1	1	4
A43	[73]	2024	Decoder	Time-series data Decomposition	1	1	1	1	4
A44	[84]	2024	Encoder	Patching, Instance Normalization, Encoding	1	1	1	0.5	3.5
A45	[97]	2023	Decoder	Time-series data Decomposition, Instance Normalization	1	1	1	1	4
A46	[88]	2023	Decoder	Instance Normalization	1	1	1	1	4
A47	[98]	2024	Encoder	Encoding	1	1	1	0.5	3.5
A48	[74]	2023	Encoder-Decoder	Trainable fully connected layer	1	1	1	1	4
A49	[99]	2023	Decoder	Instance Normalization, Encoding	1	1	1	1	4
A50	[48]	2024	Encoder-Decoder	Scaling, Quantization	1	1	1	1	4
A51	[49]	2023	Encoder	Segmentation, Scoring	1	1	1	0.5	3.5
A52	[100]	2024	Encoder	Patching, Instance Normalization	1	1	1	1	4

TABLE 8. *(Continued.)* **Appendix.**

A53	[101]	2023	Encoder	Scaling, Time-series data Decomposition	1	1	1	1	4
A54	[102]	2024	Decoder	Digit Space, Rescaling	1	1	1	1	4
A55	[70]	2024	Decoder	Digit space, Instance Normalization	1	1	1	1	4
A56	[47]	2024	Encoder-Decoder	Patching, Scaling	1	1	1	1	4
A57	[87]	2024	Encoder-Decoder	Not Clear	0.5	1	1	1	3.5
A58	[103]	2023	Encoder	Instance Normalization	1	1	1	1	4
A59	[86]	2024	Decoder	Scaling	1	1	1	1	4

ONA Overlapping Nonparametric Area.
PaLM Pathways Language Model.
PEMS Performance Evaluation and Monitoring System.
PEDS Prediction Error Density Score.
PLUTUS Predictive Learning Using Time-Series for Universal Solutions.
PSM Precipitation Supply Model.
QA Quality Assessment.
RNN Recurrent Neural Network.
RMSE Root Mean Squared Error.
ROUGH Robustness of Unsupervised Generative Handling.
Rtotal Total Error Rate.
RRSE Root Relative Squared Error.
SE Standard Error.
SMAPE Symmetric Mean Absolute Percentage Error.
SMD Secure Micro Data.
SSL Secure Socket Layer.
SWAT Secure Water Analysis Tool.
TEMPO Time-Series Embedding with Prompt Optimization.
TPU Tensor Processing Unit.
TSLANET Time-Series Lightweight Adaptive Network.
UAE United Arab Emirates.
UCI-HAR University of California Irvine Human Activity Recognition.
UNF Uncertainty Forecasting.
WAPE Weighted Absolute Percentage Error.
WISDM Wireless Sensor Data Mining.
WQL Weighted Quantile Loss.

DATA AVAILABILITY

The data used to conduct this SLR is available via this link: https://github.com/shamsua/SLR/

APPENDIX

See table 8.

REFERENCES

[1] P. B. Weerakody, K. W. Wong, G. Wang, and W. Ela, "A review of irregular time series data handling with gated recurrent neural networks," *Neurocomputing*, vol. 441, pp. 161–178, Jun. 2021.
[2] V. Papastefanopoulos, P. Linardatos, T. Panagiotakopoulos, and S. Kotsiantis, "Multivariate time-series forecasting: A review of deep learning methods in Internet of Things applications to smart cities," *Smart Cities*, vol. 6, no. 5, pp. 2519–2552, Sep. 2023.
[3] D. Koutsoyiannis and A. Montanari, "Statistical analysis of hydroclimatic time series: Uncertainty and insights," *Water Resour. Res.*, vol. 43, no. 5, pp. 1–12, May 2007.
[4] X. Liu and W. Wang, "Deep time series forecasting models: A comprehensive survey," *Mathematics*, vol. 12, no. 10, p. 1504, May 2024.
[5] G. Tong, Z. Ge, and D. Peng, "RSMformer: An efficient multiscale transformer-based framework for long sequence time-series forecasting," *Appl. Intell.*, vol. 54, no. 2, pp. 1275–1296, 2024.
[6] S. Kukreja, T. Kumar, A. Purohit, A. Dasgupta, and D. Guha, "A literature survey on open source large language models," in *Proc. 7th Int. Conf. Comput. Manage. Bus.*, Jan. 2024, pp. 133–143.
[7] M. A. K. Raiaan, M. S. H. Mukta, K. Fatema, N. M. Fahad, S. Sakib, M. M. J. Mim, J. Ahmad, M. E. Ali, and S. Azam, "A review on large language models: Architectures, applications, taxonomies, open issues and challenges," *IEEE Access*, vol. 12, pp. 26839–26874, 2024.
[8] N. Karanikolas, E. Manga, N. Samaridi, E. Tousidou, and M. Vassilakopoulos, "Large language models versus natural language understanding and generation," in *Proc. 27th Pan-Hellenic Conf. Prog. Comput. Informat.*, Nov. 2023, pp. 278–290.
[9] J. Su, C. Jiang, X. Jin, Y. Qiao, T. Xiao, H. Ma, R. Wei, Z. Jing, J. Xu, and J. Lin, "Large language models for forecasting and anomaly detection: A systematic literature review," 2024, *arXiv:2402.10350*.
[10] M. Jin, S. Wang, L. Ma, Z. Chu, J. Y. Zhang, X. Shi, P.-Y. Chen, Y. Liang, Y.-F. Li, S. Pan, and Q. Wen, "Time-LLM: Time series forecasting by reprogramming large language models," 2023, *arXiv:2310.01728*.
[11] J. Ye, W. Zhang, K. Yi, Y. Yu, Z. Li, J. Li, and F. Tsung, "A survey of time series foundation models: Generalizing time series representation with large language model," 2024, *arXiv:2405.02358*.
[12] S. S. Srinivas, C. Ravuru, G. Sannidhi, and V. Runkana, "Reprogramming foundational large language models(LLMs) for enterprise adoption for spatio-temporal forecasting applications: Unveiling a new era in copilot-guided cross-modal time series representation learning," 2024, *arXiv:2408.14387*.
[13] W. Chow, L. Gardiner, H. T. Hallgrímsson, M. A. Xu, and S. Y. Ren, "Towards time series reasoning with LLMs," 2024, *arXiv:2409.11376*.
[14] M. Jin, Q. Wen, Y. Liang, C. Zhang, S. Xue, X. Wang, J. Zhang, Y. Wang, H. Chen, X. Li, S. Pan, V. S. Tseng, Y. Zheng, L. Chen, and H. Xiong, "Large models for time series and spatio-temporal data: A survey and outlook," 2023, *arXiv:2310.10196*.
[15] Y. Liang, H. Wen, Y. Nie, Y. Jiang, M. Jin, D. Song, S. Pan, and Q. Wen, "Foundation models for time series analysis: A tutorial and survey," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2024, pp. 6555–6565.
[16] X. Zhang, R. R. Chowdhury, R. K. Gupta, and J. Shang, "Large language models for time series: A survey," 2024, *arXiv:2402.01801*.
[17] L. Su, X. Zuo, R. Li, X. Wang, H. Zhao, and B. Huang, "A systematic review for transformer-based long-term series forecasting," 2023, *arXiv:2310.20218*.
[18] Y. Jiang, Z. Pan, X. Zhang, S. Garg, A. Schneider, Y. Nevmyvaka, and D. Song, "Empowering time series analysis with large language models: A survey," 2024, *arXiv:2402.03182*.
[19] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao, "Large language models: A survey," 2024, *arXiv:2402.06196*.
[20] Q. Ding, D. Ding, Y. Wang, C. Guan, and B. Ding, "Unraveling the landscape of large language models: A systematic review and future perspectives," *J. Electron. Bus. Digit. Econ.*, vol. 3, no. 1, pp. 3–19, Feb. 2024.
[21] J. Yang, H. Jin, R. Tang, X. Han, Q. Feng, H. Jiang, S. Zhong, B. Yin, and X. Hu, "Harnessing the power of LLMs in practice: A survey on ChatGPT and beyond," *ACM Trans. Knowl. Discovery Data*, vol. 18, no. 6, pp. 1–32, Jul. 2024.

[22] M. Xu, W. Yin, D. Cai, R. Yi, D. Xu, Q. Wang, B. Wu, Y. Zhao, C. Yang, S. Wang, Q. Zhang, Z. Lu, L. Zhang, S. Wang, Y. Li, Y. Liu, X. Jin, and X. Liu, "A survey of resource-efficient LLM and multimodal foundation models," 2024, *arXiv:2401.08092*.

[23] W. Zeng et al., "PanGu-α: Large-scale autoregressive pretrained Chinese language models with auto-parallel computation," 2021, *arXiv:2104.12369*.

[24] S. Lu, I. Bigoulaeva, R. Sachdeva, H. T. Madabushi, and I. Gurevych, "Are emergent abilities in large language models just in-context learning?" 2023, *arXiv:2309.01809*.

[25] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," *Renew. Sustain. Energy Rev.*, vol. 74, pp. 902–924, Jul. 2017.

[26] A. R. S. Parmezan, V. M. Souza, and G. E. Batista, "Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model," *Inf. Sci.*, vol. 484, pp. 302–337, May 2019.

[27] S. Ahmed, I. E. Nielsen, A. Tripathi, S. Siddiqui, G. Rasool, and R. P. Ramachandran, "Transformers in time-series analysis: A tutorial," 2022, *arXiv:2205.01138*.

[28] M. Waqas and U. W. Humphries, "A critical review of RNN and LSTM variants in hydrological time series predictions," *MethodsX*, vol. 13, Dec. 2024, Art. no. 102946.

[29] M. Waqas, U. W. Humphries, A. Wangwongchai, P. Dechpichai, and S. Ahmad, "Potential of artificial intelligence-based techniques for rainfall forecasting in thailand: A comprehensive review," *Water*, vol. 15, no. 16, p. 2979, Aug. 2023.

[30] Q. Guo, Z. He, Z. Wang, S. Qiao, J. Zhu, and J. Chen, "A performance comparison study on climate prediction in Weifang City using different deep learning models," *Water*, vol. 16, no. 19, p. 2870, Oct. 2024.

[31] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, "Deep learning for time series forecasting: A survey," *Big Data*, vol. 9, no. 1, pp. 3–21, Feb. 2021.

[32] V. Ekambaram, A. Jati, P. Dayama, S. Mukherjee, N. H. Nguyen, W. M. Gifford, C. Reddy, and J. Kalagnanam, "Tiny time mixers (TTMs): Fast pre-trained models for enhanced zero/few-shot forecasting of multivariate time series," 2024, *arXiv:2401.03955*.

[33] N. Chan, F. Parker, W. Bennett, T. Wu, M. Yao Jia, J. Fackler, and K. Ghobadi, "MedTsLLM: Leveraging LLMs for multimodal medical time series analysis," 2024, *arXiv:2408.07773*.

[34] D. Cao, F. Jia, S. O. Arik, T. Pfister, Y. Zheng, W. Ye, and Y. Liu, "TEMPO: Prompt-based generative pre-trained transformer for time series forecasting," 2023, *arXiv:2310.04948*.

[35] M. Cheng, Y. Chen, Q. Liu, Z. Liu, and Y. Luo, "Advancing time series classification with multimodal language modeling," 2024, *arXiv:2403.12371*.

[36] C. Liu, S. He, Q. Zhou, S. Li, and W. Meng, "Large language model guided knowledge distillation for time series anomaly detection," 2024, *arXiv:2401.15123*.

[37] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering—A systematic literature review," *Inf. Softw. Technol.*, vol. 51, no. 1, pp. 7–15, 2009.

[38] C. Wohlin, E. Mendes, K. R. Felizardo, and M. Kalinowski, "Guidelines for the search strategy to update systematic literature reviews in software engineering," *Inf. Softw. Technol.*, vol. 127, Nov. 2020, Art. no. 106366.

[39] S. Keele, "Guidelines for performing systematic literature reviews in software engineering," EBSE, Tech. Rep., 2007.

[40] A. Zakari, S. P. Lee, R. Abreu, B. H. Ahmed, and R. A. Rasheed, "Multiple fault localization of software programs: A systematic literature review," *Inf. Softw. Technol.*, vol. 124, Aug. 2020, Art. no. 106312.

[41] S. Abdullahi, M. A. Zayyad, N. Yusuf, L. I. Bagiwa, A. Nura, A. Zakari, and B. Dansambo, "Software requirements negotiation: A review on challenges," *Int. J. Innov. Comput.*, vol. 11, no. 1, pp. 1–6, Apr. 2021.

[42] A. Gupta, H. P. Gupta, B. Biswas, and T. Dutta, "Approaches and applications of early classification of time series: A review," *IEEE Trans. Artif. Intell.*, vol. 1, no. 1, pp. 47–61, Aug. 2020.

[43] P. Tang and X. Zhang, "MTSMAE: Masked autoencoders for multivariate time-series forecasting," in *Proc. IEEE 34th Int. Conf. Tools Artif. Intell. (ICTAI)*, Oct. 2022, pp. 982–989.

[44] A. Zakari, S. Abdullahi, N. M. Shagari, A. B. Tambawal, N. M. Shanono, J. Z. Maitama, R. A. Rasheed, A. Adamu, and S. M. Abdulrahman, "Spectrum-based fault localization techniques application on multiple-fault programs: A review," *Global J. Comput. Sci. Technol.*, vol. 20, pp. 41–48, Mar. 2020.

[45] S. Wu, X. Xiao, Q. Ding, P. Zhao, Y. Wei, and J. Huang, "Adversarial sparse transformer for time series forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, Jan. 2020, pp. 17105–17115.

[46] D. Cao, W. Ye, Y. Zhang, and Y. Liu, "TimeDiT: General-purpose diffusion transformers for time series foundation model," 2024, *arXiv:2409.02322*.

[47] Y. Xu, A. Liu, J. Hao, Z. Li, S. Meng, and G. Zhang, "PLUTUS: A well pre-trained large unified transformer can unveil financial time series regularities," 2024, *arXiv:2408.10111*.

[48] A. Grigoraş and F. Leon, "Synthetic time series generation for decision intelligence using large language models," *Mathematics*, vol. 12, no. 16, p. 2494, Aug. 2024.

[49] H. Kamarthi and B. A. Prakash, "Large pre-trained time series models for cross-domain time series analysis tasks," 2023, *arXiv:2311.11413*.

[50] K. Rasul, A. Ashok, A. R. Williams, H. Ghonia, R. Bhagwatkar, A. Khorasani, M. J. D. Bayazi, G. Adamopoulos, R. Riachi, N. Hassen, M. Biloš, S. Garg, A. Schneider, N. Chapados, A. Drouin, V. Zantedeschi, Y. Nevmyvaka, and I. Rish, "Lag-Llama: Towards foundation models for probabilistic time series forecasting," 2024, *arXiv:2310.08278*.

[51] A. Das, W. Kong, R. Sen, and Y. Zhou, "A decoder-only foundation model for time-series forecasting," 2023, *arXiv:2310.10688*.

[52] E. Eldele, M. Ragab, Z. Chen, M. Wu, and X. Li, "TSLANet: Rethinking transformers for time series representation learning," 2024, *arXiv:2404.08472*.

[53] S. Gao, T. Koker, O. Queen, T. Hartvigsen, T. Tsiligkaridis, and M. Zitnik, "UniTS: A unified multi-task time series model," 2024, *arXiv:2403.00131*.

[54] Z. Pan, Y. Jiang, S. Garg, A. Schneider, Y. Nevmyvaka, and D. Song, "S^2IP-LLM: Semantic space informed prompt learning with LLM for time series forecasting," in *Proc. 41st Int. Conf. Mach. Learn.*, 2024, pp. 1–12.

[55] C. Liu, S. Yang, Q. Xu, Z. Li, C. Long, Z. Li, and R. Zhao, "Spatial–temporal large language model for traffic prediction," 2024, *arXiv:2401.10134*.

[56] E. Fons, R. Kaur, S. Palande, Z. Zeng, T. Balch, M. Veloso, and S. Vyetrenko, "Evaluating large language models on time series feature understanding: A comprehensive taxonomy and benchmark," 2024, *arXiv:2404.16563*.

[57] R. Jin, Q. Xu, M. Wu, Y. Xu, D. Li, X. Li, and Z. Chen, "LLM-based knowledge pruning for time series data analytics on edge-computing devices," 2024, *arXiv:2406.08765*.

[58] T. Zhou, P. Niu, L. Sun, and R. Jin, "One fits all: Power general time series analysis by pretrained LM," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, 2023, pp. 43322–43355.

[59] J. Wang, G. Bai, W. Cheng, Z. Chen, L. Zhao, and H. Chen, "POND: Multi-source time series domain adaptation with information-aware prompt tuning," 2023, *arXiv:2312.12276*.

[60] Y. Bian, X. Ju, J. Li, Z. Xu, D. Cheng, and Q. Xu, "Multi-patch prediction: Adapting LLMs for time series representation learning," 2024, *arXiv:2402.04852*.

[61] J. Liu, C. Zhang, J. Qian, M. Ma, S. Qin, C. Bansal, Q. Lin, S. Rajmohan, and D. Zhang, "Large language models can deliver accurate and interpretable time series anomaly detection," 2024, *arXiv:2405.15370*.

[62] H. Du, J. Zhao, Y. Zhao, S. Xu, X. Lin, Y. Chen, L. M. Gardner, and H. F. Yang, "Advancing real-time pandemic forecasting using large language models: A COVID-19 case study," 2024, *arXiv:2404.06962*.

[63] Y. Wang, R. Jin, M. Wu, X. Li, L. Xie, and Z. Chen, "K-link: Knowledge-link graph from LLMs for enhanced representation learning in multivariate time-series data," 2024, *arXiv:2403.03645*.

[64] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 1–14.

[65] E. Sayed, M. Maher, O. Sedeek, A. Eldamaty, A. Kamel, and R. E. Shawi, "GizaML: A collaborative meta-learning based framework using LLM for automated time-series forecasting," in *Proc. EDBT*, 2024, pp. 1–4.

[66] C. Liu, Q. Xu, H. Miao, S. Yang, L. Zhang, C. Long, Z. Li, and R. Zhao, "TimeCMA: Towards LLM-empowered multivariate time series forecasting via cross-modality alignment," 2024, *arXiv:2406.01638*.

[67] A. F. Ansari, L. Stella, C. Turkmen, X. Zhang, P. Mercado, H. Shen, O. Shchur, S. S. Rangapuram, S. P. Arango, S. Kapoor, J. Zschiegner, D. C. Maddix, H. Wang, M. W. Mahoney, K. Torkkola, A. G. Wilson, M. Bohlke-Schneider, and Y. Wang, "Chronos: Learning the language of time series," 2024, *arXiv:2403.07815*.

[68] F. Jia, K. Wang, Y. Zheng, D. Cao, and Y. Liu, "GPT4MTS: Prompt-based large language model for multimodal time-series forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, Mar. 2024, pp. 23343–23351.
[69] C. Chang, W.-Y. Wang, W.-C. Peng, and T.-F. Chen, "LLM4TS: Aligning pre-trained LLMs as data-efficient time-series forecasters," 2023, *arXiv:2308.08469*.
[70] R. Kaur, Z. Zeng, T. Balch, and M. Veloso, "LETS-C: Leveraging language embedding for time series classification," 2024, *arXiv:2407.06533*.
[71] U. U. Rehman, M. Hussain, T. D. T. Nguyen, and S. Lee, "Let's hide from LLMs: An adaptive contextual privacy preservation method for time series data," in *Proc. 6th Artif. Intell. Cloud Comput. Conf. (AICCC)*, Dec. 2023, pp. 196–203.
[72] H. Liu, Z. Zhao, J. Wang, H. Kamarthi, and B. A. Prakash, "LSTPrompt: Large language models as zero-shot time series forecasters by long-short-term prompting," 2024, *arXiv:2402.16132*.
[73] Y. Liu, G. Qin, X. Huang, J. Wang, and M. Long, "AutoTimes: Autoregressive time series forecasters via large language models," 2024, *arXiv:2402.02370*.
[74] M. Zhou, F. Li, F. Zhang, J. Zheng, and Q. Ma, "Meta in-context learning: Harnessing large language models for electrical data classification," *Energies*, vol. 16, no. 18, p. 6679, Sep. 2023.
[75] V. Ceperic and T. Markovic, "Transforming time-series data for improved LLM-based forecasting through adaptive encoding," *Int. J. Simulation, Syst., Sci. Technol.*, vol. 25, no. 1, pp. 1–18, Mar. 2024.
[76] N. Gruver, M. Finzi, S. Qiu, and A. G. Wilson, "Large language models are zero-shot time series forecasters," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, Jan. 2024, pp. 1–16.
[77] B. Liu, X. Liu, S. Gao, X. Cheng, and L. Yang, "LLM4CP: Adapting large language models for channel prediction," 2024, *arXiv:2406.14440*.
[78] G. Chatzigeorgakidis, K. Lentzos, and D. Skoutas, "MultiCast: Zero-shot multivariate time series forecasting using LLMs," in *Proc. IEEE 40th Int. Conf. Data Eng. Workshops (ICDEW)*, May 2024, pp. 119–127.
[79] J. Su, S. Nair, and L. Popokh, "Leveraging large language models for VNF resource forecasting," in *Proc. IEEE 10th Int. Conf. Netw. Softwarization (NetSoft)*, Jun. 2024, pp. 258–262.
[80] X. Zhou, Q. Jia, Y. Hu, R. Xie, T. Huang, and F. R. Yu, "GenG: An LLM-based generic time series data generation approach for edge intelligence via cross-domain collaboration," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, May 2024, pp. 1–6.
[81] M. Jin, Y. Zhang, W. Chen, K. Zhang, Y. Liang, B. Yang, J. Wang, S. Pan, and Q. Wen, "Position: What can large language models tell us about time series analysis," 2024, *arXiv:2402.02713*.
[82] J. Morales-García, A. Llanes, F. Arcas-Túnez, and F. Terroso-Sáenz, "Developing time series forecasting models with generative large language models," *ACM Trans. Intell. Syst. Technol.*, 2024.
[83] M. A. Merrill, M. Tan, V. Gupta, T. Hartvigsen, and T. Althoff, "Language models still struggle to zero-shot reason about time series," 2024, *arXiv:2404.11757*.
[84] M. Tan, M. A. Merrill, V. Gupta, T. Althoff, and T. Hartvigsen, "Are language models actually useful for time series forecasting?" 2024, *arXiv:2406.16964*.
[85] T. Ogawa, K. Yoshioka, K. Fukuda, and T. Morita, "Prediction of actions and places by the time series recognition from images with multimodal LLM," in *Proc. IEEE 18th Int. Conf. Semantic Comput. (ICSC)*, Feb. 2024, pp. 294–300.
[86] T. D. P. Edwards, J. Alvey, J. Alsing, N. H. Nguyen, and B. D. Wandelt, "Scaling-laws for large time-series models," 2024, *arXiv:2405.13867*.
[87] W. Liao, F. Porte-Agel, J. Fang, C. Rehtanz, S. Wang, D. Yang, and Z. Yang, "TimeGPT in load forecasting: A large time series model perspective," 2024, *arXiv:2404.04885*.
[88] X. Yu, Z. Chen, Y. Ling, S. Dong, Z. Liu, and Y. Lu, "Temporal data meets LLM—Explainable financial time series forecasting," 2023, *arXiv:2306.11025*.
[89] H. Tang, C. Zhang, M. Jin, Q. Yu, Z. Wang, X. Jin, Y. Zhang, and M. Du, "Time series forecasting with LLMs: Understanding and enhancing model capabilities," 2024, *arXiv:2402.10835*.
[90] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Jun. 2017, pp. 5998–6008.
[91] X. Zhou, E. Yılmaz, Y. Long, Y. Li, and H. Li, "Multi-encoder–decoder transformer for code-switching speech recognition," 2020, *arXiv:2006.10414*.
[92] L. Wang, Y. He, L. Li, X. Liu, and Y. Zhao, "A novel approach to ultra-short-term multi-step wind power predictions based on encoder–decoder architecture in natural language processing," *J. Cleaner Prod.*, vol. 354, Jun. 2022, Art. no. 131723.
[93] C. Chang, W.-Y. Wang, W.-C. Peng, and T.-F. Chen, "LLM4TS: Aligning pre-trained LLMs as data-efficient time-series forecasters," Tech. Rep., 2023.
[94] J. Wang, G. Bai, W. Cheng, Z. Chen, L. Zhao, and H. Chen, "POND: Multi-source time series domain adaptation with information-aware prompt tuning," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2024, pp. 3140–3151.
[95] F. Drinkall, E. Rahimikia, J. B. Pierrehumbert, and S. Zohren, "Time machine GPT," 2024, *arXiv:2404.18543*.
[96] Y. Liu, H. Zhang, C. Li, X. Huang, J. Wang, and M. Long, "Timer: Generative pre-trained transformers are large time series models," in *Proc. 41st Int. Conf. Mach. Learn.*, 2023, pp. 1–31.
[97] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proc. AAAI Conf. Artif. Intell.*, vol. 37, Jun. 2023, pp. 11121–11128.
[98] N. M. Foumani, C. W. Tan, G. I. Webb, H. Rezatofighi, and M. Salehi, "Series2vec: Similarity-based self-supervised representation learning for time series classification," *Data Mining Knowl. Discovery*, vol. 38, no. 4, pp. 2520–2544, Jul. 2024.
[99] I. de Zarzà, J. de Curtò, G. Roig, and C. T. Calafate, "LLM multimodal traffic accident forecasting," *Sensors*, vol. 23, no. 22, p. 9225, Nov. 2023.
[100] G. Woo, C. Liu, A. Kumar, C. Xiong, S. Savarese, and D. Sahoo, "Unified training of universal time series forecasting transformers," 2024, *arXiv:2402.02592*.
[101] S. Dooley, G. S. Khurana, C. Mohapatra, S. Naidu, and C. White, "ForecastPFN: Synthetically-trained zero-shot forecasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 36, Jan. 2023, pp. 1–13.
[102] R. Cao and Q. Wang, "An evaluation of standard statistical models and LLMs on time series forecasting," 2024, *arXiv:2408.04867*.
[103] H. Xue and F. D. Salim, "PromptCast: A new prompt-based learning paradigm for time series forecasting," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 6851–6864, Nov. 2024.
[104] G. Perković, A. Drobnjak, and I. Botički, "Hallucinations in LLMs: Understanding and addressing challenges," in *Proc. 47th MIPRO ICT Electron. Conv. (MIPRO)*, May 2024, pp. 2084–2088.
[105] S. A. Habashi, M. Koyuncu, and R. Alizadehsani, "A survey of COVID-19 diagnosis using routine blood tests with the aid of artificial intelligence techniques," *Diagnostics*, vol. 13, no. 10, p. 1749, May 2023.
[106] C. D. Dao, "Incorporating sparse attention mechanism into transformer for object detection in images," Tech. Rep., 2022.
[107] S. Wang, F. Liu, and B. Liu, "Escaping the gradient vanishing: Periodic alternatives of Softmax in attention mechanism," *IEEE Access*, vol. 9, pp. 168749–168759, 2021.
[108] H. Wang, Z. Zhang, and S. Han, "SpAtten: Efficient sparse attention architecture with cascade token and head pruning," in *Proc. IEEE Int. Symp. High-Performance Comput. Archit. (HPCA)*, Feb. 2021, pp. 97–110.
[109] F. Daneshfar, A. Bartani, and P. Lotfi, "Image captioning by diffusion models: A survey," *Eng. Appl. Artif. Intell.*, vol. 138, Dec. 2024, Art. no. 109288.
[110] B. Chen, T. Dao, E. Winsor, Z. Song, A. Rudra, and C. Ré, "Scatterbrain: Unifying sparse and low-rank attention," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, Dec. 2021, pp. 17413–17426.
[111] Y. Zhang, L. Ma, S. Pal, Y. Zhang, and M. Coates, "Multi-resolution time-series transformer for long-term forecasting," in *Proc. Int. Conf. Artif. Intell. Statist.*, Jan. 2023, pp. 4222–4230.
[112] C. Shao, F. Meng, and J. Zhou, "Patch-level training for large language models," 2024, *arXiv:2407.12665*.
[113] M. Li, Z. Wu, and X. Zhang, "Tokenization strategies for time-series data: A comparative study," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 4, pp. 1234–1245, Apr. 2023.
[114] M. P. Do, Y. V. M. P. Ngo, and N. V. Q. Dinh, "Frequency-based time-series modeling with Fourier and wavelet transforms," *J. Time Ser. Anal.*, vol. 45, no. 2, pp. 103–120, 2021.
[115] S. Lin, W. Lin, W. Wu, F. Zhao, R. Mo, and H. Zhang, "SegRNN: Segment recurrent neural network for long-term time series forecasting," 2023, *arXiv:2308.11200*.
[116] Y. Dang, K. Huang, J. Huo, Y. Yan, S. Huang, D. Liu, M. Gao, J. Zhang, C. Qian, K. Wang, Y. Liu, J. Shao, H. Xiong, and X. Hu, "Explainable and interpretable multimodal large language models: A comprehensive survey," 2024, *arXiv:2412.02104*.

[117] M.-H. Guo, Z.-N. Liu, T.-J. Mu, and S.-M. Hu, "Beyond self-attention: External attention using two linear layers for visual tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5436–5447, May 2023.
[118] M. Schröder, A. Zamanian, and N. Ahmidi, "Post-hoc saliency methods fail to capture latent feature importance in time series data," in *Proc. Int. Workshop Trustworthy Mach. Learn. Healthcare*. Cham, Switzerland: Springer, Jan. 2023, pp. 106–121.
[119] S. Chaudhari, V. Mithal, G. Polatkan, and R. Ramanath, "An attentive survey of attention models," *ACM Trans. Intell. Syst. Technol.*, vol. 12, no. 5, pp. 1–32, Oct. 2021.
[120] K. Men, N. Pin, S. Lu, Q. Zhang, and H. Wang, "Large language models with novel token processing architecture: A study of the dynamic sequential transformer," Tech. Rep., 2024.
[121] R. Oelschlager, "Evaluating the impact of hallucinations on user trust and satisfaction in LLM-based systems," Tech. Rep., 2024.
[122] M. Farina, U. Ahmad, A. Taha, H. Younes, Y. Mesbah, X. Yu, and W. Pedrycz, "Sparsity in transformers: A systematic literature review," *Neurocomputing*, vol. 582, May 2024, Art. no. 127468.
[123] P. Singhal, R. Walambe, S. Ramanna, and K. Kotecha, "Domain adaptation: Challenges, methods, datasets, and applications," *IEEE Access*, vol. 11, pp. 6973–7020, 2023.
[124] Q. Wen, Y. Zhang, and X. Zhang, "Towards universal time-series representation learning via global information modelling," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2022, pp. 1–12.
[125] S. M. T. I. Tonmoy, S. M. M. Zaman, V. Jain, A. Rani, V. Rawte, A. Chadha, and A. Das, "A comprehensive survey of hallucination mitigation techniques in large language models," 2024, *arXiv:2401.01313*.

SHAMSU ABDULLAHI received the bachelor's degree in computer science from Umaru Musa Yaradua University, Katsina, Nigeria, in 2012, and the master's degree in software engineering from the University of Malaya, Malaysia, in 2020. He is currently pursuing the Ph.D. degree with the Department of Computer and Information Sciences, Universiti Teknologi PETRONAS (UTP), Malaysia. His Ph.D. research is focused on artificial intelligence, specifically on mitigating hallucinations in time-series LLMs. Since January 2015, he has been a Lecturer with the Department of Computer Science, Hassan Usman Katsina Polytechnic, Katsina State, Nigeria. He has presented at many local and international conferences and published extensively in reputable international journals. His research interests include hallucinations in LLMs, time-series LLMs, contrastive learning, software requirements engineering, software testing, and big data technologies.

KAMALUDDEEN USMAN DANYARO (Member, IEEE) received the bachelor's degree in mathematics from Bayero University, Kano, Nigeria, the master's degree in business information technology from Northumbria University, Newcastle, U.K., and the Ph.D. degree from Universiti Teknologi PETRONAS. He is currently a Senior Lecturer with the Computer and Information Science Department, Universiti Teknologi PETRONAS (UTP), Perak, Malaysia. He is also a Researcher and a member of the Centre for Cyber-Physical Systems (C2PS), Institute of Emerging Digital Technology (EDiT), UTP. His research interests include data science, artificial intelligence, computer networks, and security. He serves as a reviewer for numerous conferences and journals.

ABUBAKAR ZAKARI received the master's degree in computer networks from Middlesex University, London, in 2014, and the Ph.D. degree in software engineering from the University of Malaya, Malaysia, in 2019. His current research interests include software testing, software fault localization, dynamic software update, graph, theory, generative AI, and LLMs.

IZZATDIN ABDUL AZIZ received the Bachelor of Technology degree (Hons.) in information technology from Universiti Teknologi PETRONAS (UTP), in 2002, the master's degree in information technology from The University of Sydney, Australia, in 2004, and the Ph.D. degree in information technology from Deakin University, Australia, in 2014. He is currently an Associate Professor with the Department of Computer and Information Sciences, UTP, Malaysia, and heads the Center for Research in Data Science (CeRDaS). His work focuses on solving complex upstream and downstream oil and gas (O&G) industry challenges through machine learning and big data analytics, working closely with O&G companies to deliver solutions for issues like offshore pipeline corrosion rate prediction, pipeline corrosion detection, rotating machinery, process failure prediction, and securing cloud-based data. His research also spans bridging upstream and downstream O&G operations through data analytics and addressing fundamental computer science problems, including algorithm optimization.

NOOR AMILA WAN ABDULLAH ZAWAWI received the Bachelor of Science degree (Hons.) in housing, building and planning and the Master of Science degree in building technology from Universiti Sains Malaysia, in 1998 and 1999, respectively, and the Ph.D. degree in built environment from International Islamic University Malaysia, in 2011. She is currently the Senior Director of Technology Research Excellence (TREx) with Universiti Teknologi PETRONAS (UTP), Malaysia, where she is also an Associate Professor with the Department of Civil and Environmental Engineering. Her main research interests include urban redevelopment, building technology, decommissioning cost estimation, sustainable approach, and multiple criteria decision analysis and GIS. She is also leading research on decommissioning of offshore platform with the Offshore Engineering Centre, UTP, with a focus on construction management and sustainability.

SHAMSUDDEEN ADAMU received the B.Sc. degree in computer science from Ahmadu Bello University, Zaria, and the M.Sc. degree in information technology from the Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Malaysia, where he is currently pursuing the Ph.D. degree. He has over 11 years of academic experience. His research interests include machine learning, deep learning, data mining, databases, and data optimization.