



Large Language Models for Time Series: A Survey

☀ 상태	완료
≡ Journal	IJCAI
≡ Year	2024.05
≡ Summary	LLMs for time series analysis 적용을 위한 다양한 방법론 탐구 및 분류체계 제공
≡ Limitations	LLM의 숫자 데이터 일반화 가능성 연구, 멀티모달·멀티태스킹·도메인 최적화 부족, 시계열 의미정보 왜곡 없는 알고리즘 필요
🔗 Link	https://arxiv.org/pdf/2402.01801
≡ category	LLM Time Series survey

<https://github.com/xiyuanzh/awesome-llm-time-series>

Introduction

시계열 분석은 전통적으로 frequency analysis, decomposition-based approaches를 사용하여 분석했고 최근들어 CNN, LSTM, Transformers를 사용한 모델을 사용함. LLM과 LMM은 최근 몇 년 동안 큰 성공을 얻었고, 시계열 분석 영역에서 슬슬 사용하려는 추세임. 하지만, 텍스트 데이터와 시계열의 연속성 간 생기는 modality gap에 대해서는 연구가 부족함. 따라서, 본 논문에서는 modality gap을 연결하고 LLM for time series analysis에서 지식을 전달하는 부분에 초점을 맞춰 문헌 조사를 함.



Figure 1: Large language models have recently been applied for various time series tasks in diverse application domains.

LLM 파이프라인을 input, tokenization, embedding, LLM, output으로 구분하고 각각의 단계에 따라 카테고리화를 진행함.

1. *Prompting*(input stage)
2. *Time Series Quantization*(tokenization stage) : time series as special token
3. *Aligning*(embedding stage) : time series encoder to align time series embeddings with language space
4. *Vision as Bridge*(LLM stage) : visual representations as a bridge
5. *Tool Integration*(output stage)

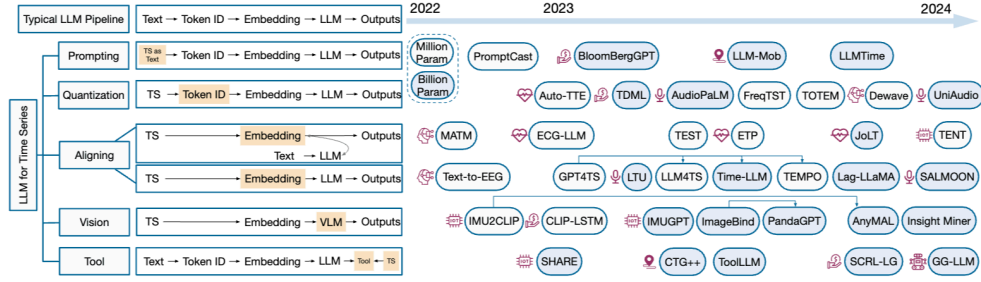


Figure 2: Left: Taxonomy of LLMs for time series analysis (prompting, quantization, aligning which is further categorized into two groups as detailed in Figure 4, vision as bridge, tool integration). Right: We present representative works for each category, sorted by their publication dates. The use of arrows indicates that later works build upon earlier studies. Dark(light)-colored boxes represent billion(million)-parameter models. Icons to the left of the text boxes represent the application domains of domain-specific models, with icons' meanings illustrated in Figure 1.

Background and Problem Formulation

LLM은 GPT-2, BERT, BART, T5와 같은 모델이 기본 아키텍처로 자리잡고, 시간이 지남에 따라 LLAMA-2, PaLM, GPT-4 등 멀티모달 대규모 언어모델이 발전하였음.

- 입력 : 시계열 $x_s \in \mathbb{R}^{T \times c}$ 및 선택적 텍스트 데이터 x_t 로 구성된 x 로 나타냄. T 는 시퀀스 길이, c 는 특성의 수를 나타냄.
- 출력 : y 로 나타내며, 특정 하위 작업에 따라 시계열, 텍스트 또는 숫자를 나타낼 수 있음. 시계열 생성 또는 예측 작업의 경우 y 는 생성된 시계열 y_s 또는 예측된 k -스텝 미래 시계열 $y_{T+1:T+k}^s$ 를 나타냄. 텍스트 생성 작업의 경우, y 는 텍스트 데이터 y_t 를 나타냄. 시계열 분류 또는 회귀 작업의 경우, y 는 예측된 클래스나 숫자 값을 나타내는 숫자를 나타냄.
- 모델 : 우리는 언어, 시계열 및 비전 모델을 나타내기 위해 각각 f_θ , g_ϕ , h_ψ 로 매개변수화함. 여기서 f_θ 는 일반적으로 pre-trained LLM에서 초기화됨. 손실 함수 L 을 통해 매개변수 θ , ϕ 및 ψ 를 최적화함.

Taxonomy

Table 2: Summary of five major categories of applying LLMs for time series analysis, including their respective subcategories, representative works, mathematical formulations, advantages and limitations. q and x_v represent text-based quantization process and image data.

Method	Subcategory	Representative Works	Equations	Advantages	Limitations
Prompting	Number-Agnostic	PromptCast [Xue and Salim, 2022]	$y = f_\theta(x_s, x_t)$	easy to implement; zero-shot capability	lose semantics; not efficient
	Number-Specific	LLMTime [Gruver <i>et al.</i> , 2023]			
Quantization	VQ-VAE	DeWave [Duan <i>et al.</i> , 2023]	$k_i = \arg \min_j \ g_\phi(x_s)_i - c_j\ _2$	flexibility of index and time series conversion	may require two-stage training
	K-Means	AudioLM [Borsos <i>et al.</i> , 2023]	$\mathbf{k} = [k_i]_{i=1}^T, \mathbf{y} = f_\theta(\mathbf{k}, x_t)$		
	Text Categories	TDML [Yu <i>et al.</i> , 2023]	$\mathbf{y} = f_\theta(q(x_s), x_t)$		
Aligning	Similarity Match	ETP [Liu <i>et al.</i> , 2023a]	$\mathbf{y} = g_\phi(x_s)$	align semantics of different modalities; end-to-end training	complicated design and fine-tuning
	LLM Backbone	MATM [Han <i>et al.</i> , 2022]	$\mathcal{L} = \text{sim}(g_\phi(x_s), f_\theta(x_t))$		
Vision as Bridge	Paired Data	GPT4TS [Zhou <i>et al.</i> , 2023a]	$\mathbf{y} = f_\theta(g_\phi(x_s), x_t)$	additional visual knowledge	not hold for all data
	TS Plots as Images	ImageBind [Girdhar <i>et al.</i> , 2023]	$\mathcal{L} = \text{sim}(g_\phi(x_s), h_\psi(x_v))$		
Tool	Code	Wimmer and Rekabsaz [2023]	$\mathbf{y} = h_\psi(x_s)$	empower LLM with more abilities	optimization not end-to-end
	API	CTG++ [Zhong <i>et al.</i> , 2023]	$z = f_\theta(x_t)$		
		ToolLLM [Qin <i>et al.</i> , 2023]	$\mathbf{y} = z(x_s)$		

Prompting

PromptCast, Liu et al. , TabLLM과 같이 Number-Agnostic Tokenization 방식이 있는 반면, LLMTime, BloombergGPT, Mirchandani et al. 과 같이 Number-Specific Tokenization 방식이 있음. 이는 각 숫자에 구분자를 넣거나, 개별 토큰으로 처리함으로써 숫자를 보다 잘 처리함.

Table 1: Examples of representative direct prompting methods.

Method	Example
PromptCast [Xue and Salim, 2022]	“From $\{t_1\}$ to $\{t_{\text{obs}}\}$, the average temperature of region $\{U_m\}$ was $\{x_t^m\}$ degree on each day. What is the temperature going to be on $\{t_{\text{obs}}\}$?”
Liu <i>et al.</i> [2023d]	“Classify the following accelerometer data in meters per second squared as either walking or running: 0.052,0.052,0.052,0.051,0.052,0.055,0.051,0.056,0.06,0.064”
TabLLM [Hegselmann <i>et al.</i> , 2023]	“The person is 42 years old and has a Master’s degree. She gained \$594. Does this person earn more than 50000 dollars? Yes or no? Answer:”
LLMTime [Gruver <i>et al.</i> , 2023]	“0.123, 1.23, 12.3, 123.0” \rightarrow “1 2, 1 2 3, 1 2 3 0, 1 2 3 0 0”

Quantization

수치 데이터를 LLM의 입력으로 사용할 수 있도록 numerical data \rightarrow discrete representations로 바꿈. discrete representations 기술에 따라 2가지로 나뉨. (a) VQ-VAE(Vector Quantized-Variational AutoEncoder)는 코드북을 학습하여 인코딩된 time series representations($g_\phi(x_s) \in R^{TS \times D}$)의 각 단계 i 의 가장 가까운 이웃 k_i 를 코드북에서 식별하고, 이에 해당하는 인덱스 k 를 토큰화된 입력으로 사용함. Auto-TTE(ECG), DeWave(EEG-to-text translation task), TOTEM, UniAudio, VioLA, AudioGen(vector quantization layers 사용) 등 다양한 모델이 있음.

(b) Discrete Indices from K-Means는 중심 인덱스를 이산화된 토큰으로 사용하는 방법임. 대부분 오디오 영역에서 많이 사용함. SpeechGPT, AudioLM, AudioPaLM(PaLM-2 + AudioLM)이 있음.

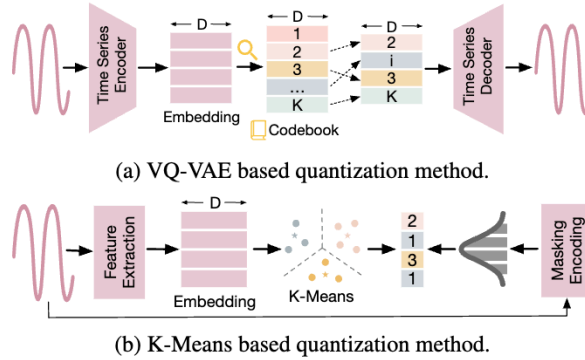


Figure 3: Two types of index-based quantization methods.

(c) 다른 양자화 기법도 존재함. ReqTST의 경우 주파수 스펙트럼을 사용하여 주파수 단위로 이산화 함. Chronos는 실수 시계열을 이산 구간으로 양자화 후 기존 언어 모델에 손실 함수를 통해 최적화 함. TDML의 경우 금융 분야에서 숫자 데이터를 미리 정의된 텍스트 범주로 변환하여 사용함. (ex. 주간 가격 변동을 D_i 또는 U_i 로 표시하여 12개 구간으로 분리)

Aligning

시계열에 대한 별도의 인코더를 훈련시키고, 인코딩된 시계열을 언어 모델의 semantic space에 정렬함. 정렬 전략에 따라 2가지로 나뉨. (a-1) Similarity Matching through Contrastive Loss. Contrastive loss를 최소화하여 시계열 임베딩과 텍스트 임베딩을 유사성 매칭을 통해 정렬함. ETP(심전도 신호를 텍스트 보고서와 매칭), King et al. (임상 측정치를 임상 메모와 매칭), TEST, TENT, JoLT(Querying Transformer 사용) 모델 등이 해당 방식 사용함.

$$L = -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(\text{sim}(g_\phi(x_{si}), f_\theta(x_{ti})))}{\sum_{k=1}^P \exp(\text{sim}(g_\phi(x_{si}), f_\theta(x_{tk})))}$$

(a-2) Similarity Matching through Other Loss. ECG-LLM(심전도와 언어 임베딩 간 분포를 Optimal Transport 기반 손실 함수 사용하여 정렬 \rightarrow ECG 보고서 생성 모델 훈련), MTAM(Canonical Correlation Analysis 및 Wasserstein Distance와 같은 다양한 정렬 기술을 손실 함수로 사용 \rightarrow EEG 특징을 해당 언어 설명과 정렬) 등 다양한 손실 함수를 적용한 모델이 있음.

(b) LLMs as Backbones. 시계열 임베딩 layer 뒤에 LLM을 백본으로 직접 사용할 수 있음. EEG-to-Text의 경우, EEG 임베딩을 pre-trained BART에 입력하여 EEG-to-text 디코딩 및 EEG 기반 감정 분류를 수행함. GPT4TS는 frozen pre-trained GPT-2를 통해 다양한 태스크에서 활용 가능한 프레임워크 제공함. 이 외에도 TEMPO(seasonal-trend decomposition), LLM4TS(2단계 fine-tuning), UniTime(도메인 설명), GATGPT(graph attention mechanism), ST-LLM, Time-LLM(시계열을 text로 재프로그래밍), Lag-Llama, WavPrompt, Speech LLaMA, MU-LLaMA, LTU, SALMONN 등이 있음.

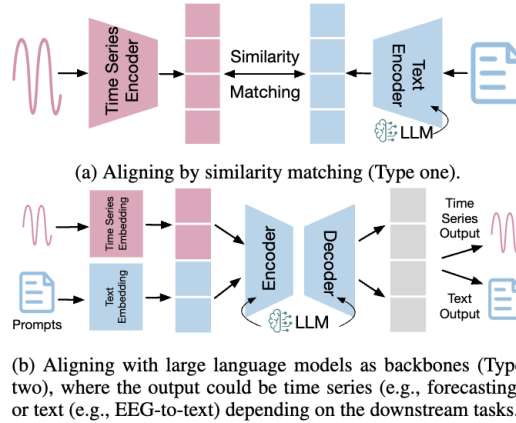


Figure 4: Two types of aligning based methods.

Vision as Bridge

시계열 데이터는 시각적으로 표현될 수 있으며, 이는 텍스트 데이터와 더 가까운 정합성을 가짐. 따라서, vision modality를 시계열과 LLM을 연결하는 브리지로 활용함. ImageBind, PandaGPT, IMU2CLIP, AnyMAL 모두 공동 임베딩 공간을 학습하여 multimodal 인코더를 만들. IMUGPT, T2M-GPT는 데이터를 생성하는 모델임. CLIP-LSTM, InsightMiner는 시계열을 라인 플롯 이미지로 변환하여 태스크를 위한 VLM의 input으로 넣음.

Tool

LLM에서 시계열을 직접적으로 처리하지 않고 API 호출 등과 같은 간접적인 도구로 시계열 태스크 처리함. CTG++, ToolLLM, SHARE, GG-LLM(LLaMA-2를 활용해 일반 인간 행동 패턴에 대한 world knowledge를 인코딩하여 추가 train 없이 인간 행동 예측), SCRL-LG 등이 있음.

Comparison within the Taxonomy

- Data
 - 훈련 데이터가 없고 zero-shot 방식으로 적용하는 경우 적합함. → but, 숫자를 문자열로 표현하기 때문에 숫자의 본질적인 의미가 감소함. → 따라서, 훈련 데이터가 있으면 quantization 혹은 aligning 방식 추천함.
- Model
 - prompting : billion-parameter 사용함.
 - quantization/aligning : 요구 사항 및 컴퓨팅 자원에 따라 million ~ billion 선택해서 사용함.
- Efficiency
 - prompting : 긴 시퀀스에서는 비효율적. 간단한 숫자 데이터를 텍스트 정보와 엮어서 다룰 때 효율적임.
 - quantization/aligning : 긴 시퀀스를 처리하기에 효율적임.
- Optimization
 - quantization : 2-stage training을 할 수 있지만 최적화되지는 않은 성능일 수 있음.
 - aligning : end-to-end training을 하기 때문에 최적화 성능은 좋음.

Multimodal Datasets

[[Zuco 1.0](#)], [[Zuco 2.0](#)]

Table 3: Summary of representative time series and text multimodal datasets.

Domain	Dataset	Size	Major Modalities	Task
Internet of Things	Ego4D ² [Grauman <i>et al.</i> , 2022]	3, 670h data, 3.85M narrations	text, IMU, video, audio, 3D	classification, forecasting
	DeepSQA ³ [Xing <i>et al.</i> , 2021]	25h data, 91K questions	text, imu	classification, question answering
Finance	PIXIU ⁴ [Xie <i>et al.</i> , 2023b]	136K instruction data	text, tables	5 NLP tasks, forecasting
	MoAT ⁵ [Lee <i>et al.</i> , 2023]	6 datasets, 2K timesteps in total	text, time series	forecasting
Healthcare	Zuco 2.0 ⁶ [Hollenstein <i>et al.</i> , 2019]	739 sentences	text, eye-tracking, EEG	classification, text generation
	PTB-XL ⁷ [Wagner <i>et al.</i> , 2020]	60h data, 71 unique statements	text, ECG	classification
	ECG-QA ⁸ [Oh <i>et al.</i> , 2023]	70 question templates	text, ECG	classification, question answering
Audio	OpenQA-5M ⁹ [Gong <i>et al.</i> , 2023]	5.6M (audio, question, answer) tuples	text, audio	tagging, classification
Music	MusicCaps ¹⁰ [Agostinelli <i>et al.</i> , 2023]	5.5K music clips	text, music	captioning, generation
Speech	CommonVoice ¹¹ [Ardila <i>et al.</i> , 2019]	7, 335 speech hours in 60 languages	text, speech	ASR, translation

Challenges and Future Directions

Theoretical Understanding

LIFT는 언어 모델의 fine-tuning이 아키텍처나 손실 함수를 변경하지 않고 비언어 작업에 효과적일 수 있음을 보여줌. GPT4TS는 LLM의 일반적인 능력이 주성분 분석(PCA)과 관련이 있을 수 있다고 주장함. self-attention layer에 대한 기밀기를 최소화하는 것이 PCA와 유사한 점을 공유하기 때문임. LLM의 숫자 데이터에 대한 일반화 가능성에 대한 추가 조사가 필수적임.

Multimodal and Multitask Analysis

Unified-IO 및 UniAudio와 같은 모델이 multiple input modalities를 분리된 어휘 토큰의 시퀀스로 통합하여 단일 Transformer 기반 아키텍처 내에서 multitask를 지원함. LLM을 multimodal 및 multitask 분석에 활용하기 위한 연구가 더욱 진행된다면 보다 강력한 시계열 기본 모델을 개발할 수 있음.

Efficient Algorithms

Patching [Nieet *al.*, 2022]은 성능 개선과 복잡성 감소를 위해 널리 채택된 전략이지만, 큰 패치가 시계열의 의미 정보를 왜곡할 수 있어 성능에 부정적인 영향을 미칠 수 있음. 따라서 대규모 시계열 분석을 촉진하고 최종 사용자와의 상호작용을 향상시키기 위해 더 효율적인 알고리즘 개발이 중요함.

Combining Domain Knowledge

기존 통계적 도메인 지식을 LLM과 결합하면 시계열 분석을 위한 모델의 성능을 향상시킬 수 있음. TEMPO는 시계열의 seasonal-trend decomposition을 적용하고 이를 pre-trained transformer의 입력함으로써 서로 다른 의미적 bias로 취급함. FreqTST는 단일 시계열을 주파수 단위로 토큰화하여 주파수 도메인에서 강점이 있음. 따라서, wavelet decomposition, auto-correlation analysis, empirical mode decomposition 등과 같이 도메인 지식을 통합하며 LLM의 시계열 분석 능력을 강화할 수 있음.

Customization and Privacy

FedAlign은 연합 학습 프레임워크를 활용하여 다양한 클라이언트 간의 latent space를 정렬하기 위한 공통 기반으로 표현적인 자연어 클래스 이름을 사용함. 모델 맞춤화 및 사용자 개인 정보 보호를 위한 연구를 발전시키는 것은 LLM 기반 시간 시계열 분석의 범위와 유용성을 넓힐 수 있음.

Conclusion

LLM의 numerical time series analysis를 위한 지식을 전달하는 분류를 체계적으로 분석함. prompting, quantization, aligning, vision modality, tool로 분류하고 이 범주에 따라 수학적 공식화 및 대표 모델, 장점과 한계를 비교함. 다양한 도메인에서 데이터셋을 소개하고 미래 연구를 위한 도전 과제를 소개함.