Generalized Prompt Tuning: How to Use a Frozen Pre-Trained Univariate Time Series Foundation Model for Multivariate Time Series Prediction

Carnegie Mellon University

Mingzhu Liu [1] Angela H. Chen [1] George H. Chen [2]

[1] CMU School of Computer Science, [2] CMU Heinz College of Information Systems and Public Policy

Introduction

🧑‍💼 **Q: How do you adapt univariate time series foundation models to your data?**

🧑‍💻 **A:** Just **fine-tune it like LLMs,** for example:

- Full fine-tuning
- LoRA
- Linear probing
- Prompt tuning (P-Tuning v2)

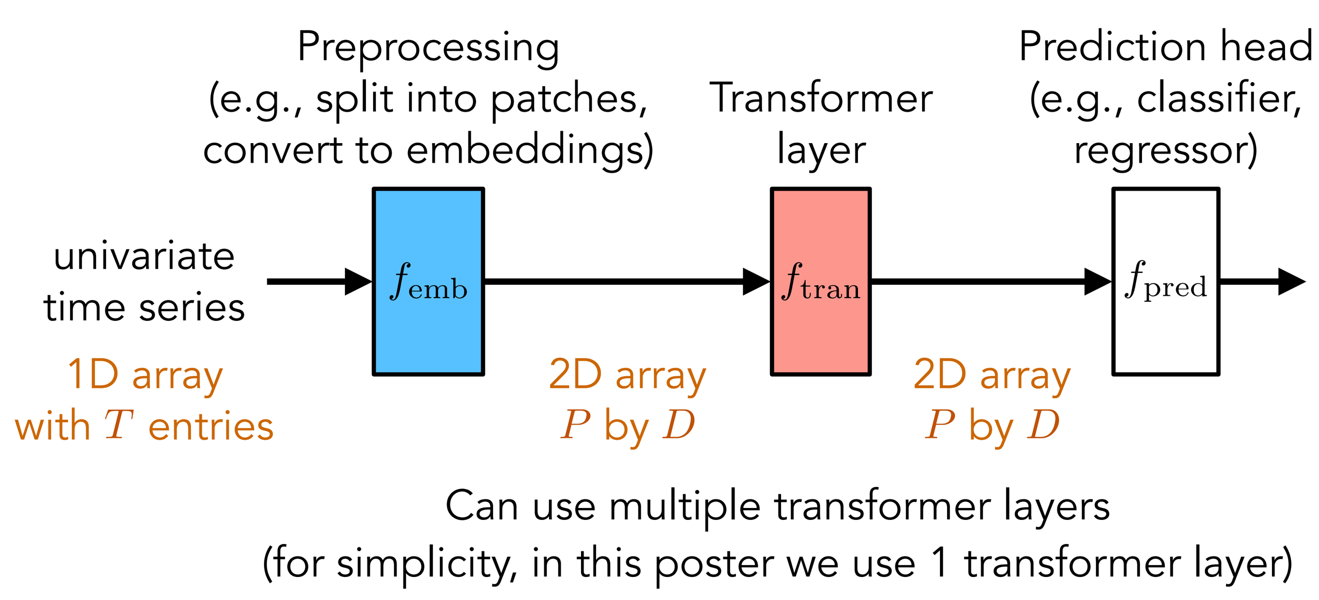
Or try our fine-tuning strategy for **multivariate time series** that we call **Generalized Prompt Tuning (Gen-P-Tuning)**

Contributions

- First benchmark on fine-tuning univariate time series foundation models on multivariate time series tasks
  - 8 **classification** datasets
    - 6 from the UEA archive
    - 2 from MIMIC-III
  - 4 **forecasting** datasets from Autoformer
- We propose Gen-P-Tuning
  - Parameter-efficient
  - Performs well in practice
  - Special cases: linear probing, standard prompt tuning

Background

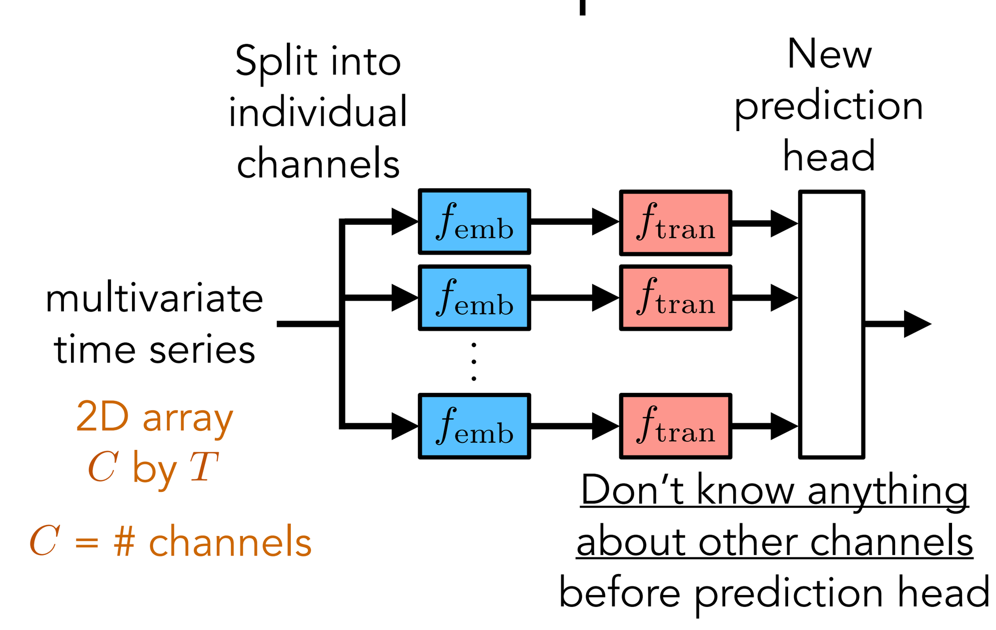
Pre-trained univariate foundation model:



T = # time steps
P = # patches
D = embedding dimension

Example models: MOMENT, GPT4TS, …

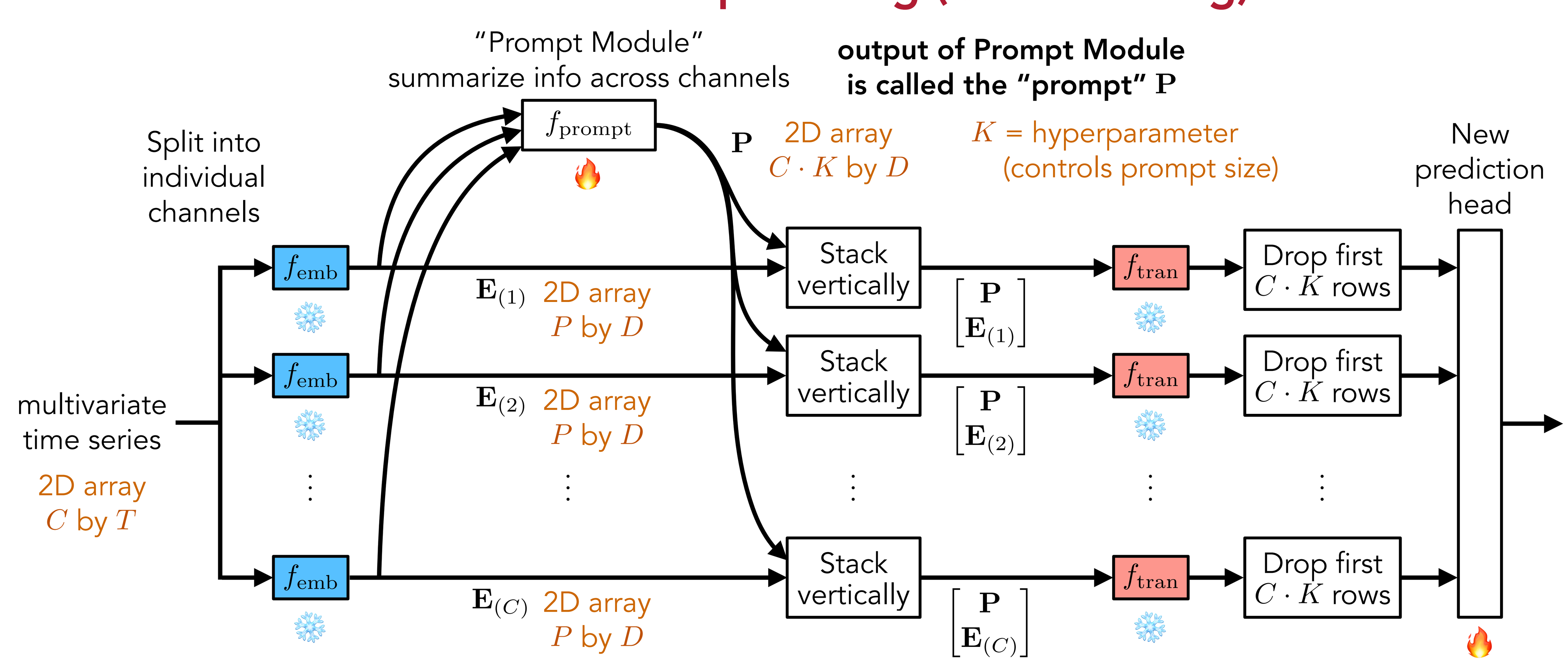
Handle multivariate time series with "channel independence":



C = # channels

We call this **"linear probing"**

Generalized Prompt Tuning (Gen-P-Tuning)



Behaves like adding $C \cdot K$ initial fictitious patches as a prefix

- The prompt $\mathbf{P}$ is a **summary across channels** that we attach as a prefix to each channel's embedding representation
- The prompt is learned through **Prompt Module** f_{prompt}
  - Can be any neural net for sequential data (eg., transformer, RNN, …)
  - If Prompt Module does not depend on inputs: get standard Prompt Tuning
  - If prompt size hyperparameter $K = 0$: get standard linear probing
- Only the Prompt Module and prediction head need to be trained 🔥

Experimental Results

For more detailed results on each dataset, see the paper

Average classification accuracy

Model	Full	LoRA	Linear Probing	Prompt Tuning	Gen-P-Tuning
MOMENT	0.732 ± 0.217	0.763 ± 0.206	**0.769** ± 0.204	0.756 ± 0.205	0.761 ± 0.207
GPT4TS	0.741 ± 0.213	**0.742** ± 0.204	**0.742** ± 0.202	0.609 ± 0.221	0.590 ± 0.243

Average forecasting mean squared error

Model	Full	LoRA	Linear Probing	Prompt Tuning	Gen-P-Tuning
MOMENT	1.211 ± 1.161	1.121 ± 1.152	3.252 ± 1.965	**1.071** ± 1.068	1.118 ± 1.137
GPT4TS	1.612 ± 1.313	1.546 ± 1.336	3.242 ± 1.757	**1.478** ± 1.255	1.482 ± 1.251

In these tables, Gen-P-Tuning uses a nontrivial Prompt Module

Findings

A longer version of this paper specialized to healthcare data has been published at Machine Learning for Health (ML4H) proceedings track

- Gen-P-Tuning is often among the best-performing ones
- Linear probing sometimes perform well, suggesting that channel-independence is sometimes sufficient
- Keep in mind that Gen-P-Tuning includes linear probing and Prompt Tuning as special cases: can choose between special cases by tuning on a validation set

This work was supported in part by NSF CAREER award #2047981