



Google Summer of Code



SKTIME

GSOC Proposal with Sktime

Personal **Details**

Name	Muhammad Armaghan Shakir
Github Username	geetu040
Email	raoarmaghanshakir040@gmail.com
Contact	+92 302 1505781
LinkedIn	Armaghan Shakir
Nationality	Pakistani
Degree	Bachelors in Software Engineering (3rd Year)
College	National University of Science and Technology, Islamabad
Timezone	UTC+05:00

Project **Details**

Organisation	Sktime : A unified framework for ML with time series
Project Title	scaling, backends, foundation models - polars, pytorch, huggingface
Project Sub Domain	Sktime integration with deep learning backends - pytorch and huggingface
Mentors	fkiraly , benHeid
Difficulty	Hard
Length	350 hours

About **Me**

I am **Armaghan**, 3rd year undergraduate student, pursuing my bachelors degree in software engineering from National University of Science and Technology, Islamabad.

I started my open-source journey by creating my first PR 🦋 [feat: Expose Setting for GRPC Channel-Level Compression at Client Side #480](#) on 04 Feb 2024. Then I found [sktime](#) and have since been making small to medium [contributions](#) there.

I call it a journey because I believe these are my first steps to a long way towards open-source.

My Experience

Research Internship

Under the [National Centre of Artificial Intelligence](#) I completed a 3 months internship in the [Deep Learning Lab](#) ([certificate](#)) and later continued with [TUKL Research and Development Lab](#).

Related Projects

- [EEG Pathology Detection](#): Working on time-series EEG dataset to detect pathology.
- [AI and Forest Fire Mitigation](#): Research on AI solutions for mitigating fires.
- [Markaz App](#): Data Analysis and Processing for an Ecommerce app.
- [Tree Counting Algorithm](#): Government Project to count trees from satellite image of Islamabad Margalla Hills.

Fiverr Freelancing

I have also created a successful [portfolio](#) on fiverr, here is my [profile](#).

Related Projects

- [FMConnect.AI](#) Hosted RAG Application using ChatGPT, Pinecone, GCP
- [Audio Transcription and Diarization](#)
- [Text to Video convertor](#)
- [Stock Prediction](#)

Personal Projects

- [pymodels](#): ML/DL models implemented from scratch using scikit-learn, pytorch and tensorflow
- [visual-dsa](#): visualising trees, graphs and sorting algorithms in react.js
- [intelli-snake](#): training snake using CNNs to avoid obstacles and reach food
- [Kaggle](#): Notebooks on EDA, Recommendation systems, ASR, Text Classification

I also have a bunch of [other projects](#) involving web development, cloud computing and artificial intelligence that can be found on my [github](#).

Throughout my academic journey, I also completed a bunch of notable courses on [coursera](#), here are my [certificates](#).

Contributions to **Sktime**

List of all my [contributions\(15\)](#) from Feb 27, 2024 onwards

- 9 Pull Requests
- 5 Issues
- 1 Code Review

Type	Title	Tags
	[MNT] change cycle for renaming cINNNForecaster to CINNNForecaster #6120	maintenance forecasting
	[ENH] cINNNForecaster needs some informative error handling and improved documentation #6158	enhancement bug doc
	[ENH] re-interface API for LTSF algorithms #6192	enhancement forecasting
	[BUG] error while fitting on cINNNForecaster #6122	forecasting bug
	[ENH] Update MSTL import in sktime.transformations.series.detrend #6085	enhancement transformations
	[ENH] Implements LTSFTransformer #6202	enhancement forecasting
	[ENH] Update doc and behaviour of freq="auto" in neuralforecast #6237	enhancement forecasting
	[MNT] change cycle (0.29.0) for renaming cINNNForecaster to CINNNForecaster #6238	maintenance

	[ENH] NeuralForecastRNN should auto-detect freq #6039	enhancement forecasting
	[MNT] initialise change cycle (0.28.0) for renaming cINNForecaster to CINNForecaster #6121	maintenance
	[MNT] create build tool to check invalid backticks #6088	documentation enhancement maintenance
	[ENH] Adds MSTL import statement in detrend #6116	-
	[ENH] Examples for YtoX transformer docstring #6059	documentation transformations
	[DOC] Fix invalid use of single-grave in docstrings #6023	documentation
	Add a test_mstl module checking if transform returns desired components #6084	enhancements transformations

Project **Idea**

Title

"Sktime integration with deep learning backends - pytorch and huggingface"

Problem Description

Sktime has interfaced a great deal of forecasters and modules while applying a consistent API design. There are still a number of deep learning forecasters and backends that are yet to a part.

In the domain of deep learning there are 3 main areas to work on:

Pytorch Adapter

A [pytorch adapter](#) already exists that was originally implemented by [Luca-miniati](#) in this [commit](#). Although this adapter has been improved, yet it comes with some limitations. The adapter needs some enhancements and support with design, enabling cuda and a bunch of forecasters that can be added on top of this adapter.

Pytorch-forecasting

Pytorch-forecasting library has a variety of forecasting models. Interfacing this library with sktime, would expand the general coverage of forecasting models in sktime.

Huggingface Interface

Transformers with their complex architectures, pre-trained models and fine-tuning options, makes huggingface an interesting choice to be integrated with sktime.

Deliverables

1. pytorch adapter for forecasting and classification module with forecasting algorithms implemented.
2. pytorch-forecasting adapter and forecasting models on top of it.
3. huggingface interface

Work Plan

Pre-GSOC Contribution (- 30 Apr)

1. Initial Contributions (23 Feb - 15 March)
2. GSoC Draft Proposal (15 Mar - 20 Mar)
3. [Mid Semester Exams](#) (22 Mar - 29 Mar)
4. GSoC Final Proposal (30 Mar - 2 Apr)
5. More Contributions (2 Apr - 30 Apr)
 - a. Complete LTSFTransformer [PR#6202](#).
 - b. Fix auto freq errors in neuralforecast [PR#6237](#).

Community Bonding Period (*1 May - 26 May*)

1. Bond with the mentors and other contributors
2. Learn more about the project
3. Work on small to medium issues relevant to the project; implement OS-CNN [#5828](#) on top of pytorch-adapter.
4. [End Semester Exams](#) (*24 May - 31 May*)

Phase - 1 (*27 May - 8 July*)

Week 1 (27 May - 3 June)

- [End Semester Exams](#) (*24 May - 31 May*)
- Flight to Germany
- (compensation discussed in [availability](#))

Week 2 (3 June - 10 June)

1. Enhance existing pytorch-adapter
 - a. Support for exogenous data
 - i. Current adapter does not support exogenous data. This support can and must be extended as it is already supported by the same algorithms implemented by [neuralforecast](#), specifically I am talking about LTSF algorithms.
 - ii. Although the Dataset class seems to use exogenous data, the X provided during fit is not used ahead.
 - iii. This can be fixed by passing this X to the Dataset class.
 - iv. This might not work with the existing Dataset class design which should be appraised and fixed.
 - b. Support for pytorch Schedulers in training and Samplers in Dataset.
 - i. Add them as init parameters and use where needed in training and dataset initialization.
 - c. Cuda support for Dataset and model training

- i. Take cuda option as init parameter
- ii. Instantiate dataset and model on device specified by cuda option.
- iii. Assure valid conversions of sktime interface data types to device specific tensors during fit and predict.

Week 3 (10 June – 17 June)

2. Support for transformers networks in pytorch-adapter

- a. Training Transformer networks take more than just x in a forward pass. I stumbled upon this problem while [implementing the LTSF-Transformer](#) and realised the current implementation cannot train attention based architectures.
- b. This requires some changes in the base adapter design.
 - i. We can create another adapter on top of the existing adapter for these cases.
 - ii. Or we can enhance the pytorch base adapter's capability to handle different types of architectures (one forward pass designed for all architectures)
- c. Though my preference is to go with (i) yet this decision is to be made through moot with mentors.

Week 4 (17 June – 24 June)

3. Pytorch Adapter for classification module

- a. Copy the pytorch adapter from the forecasting module.
- b. Make changes to the forward pass to make it a classification problem.
- c. Change default Loss
- d. And I hope with testing and minor changes, we will have the adapter implemented.

Week 5 (24 June - 1 July) + Week 6 (1 July - 8 July)

4. Implement Pytorch based forecasters

- a. I once worked on [predicting stock prices](#), where I implemented a deep neural network using pytorch from scratch and it completed the job for me. We can create similar architectures, native to sktime, that can be used for testing and comparing with other approaches.
- b. Or we can implement one of the architectures stated here: [issues#4691\(comment\)](#) and [Workstream: Deep Learning based Forecasters](#) and [issue#6177](#)
- c. Choose a forecasting algorithm from the provided list after discussing with mentors.
- d. Read the original paper associated with it for better understanding of the algorithm.
- e. Search for existing implementation of the algorithm in pytorch.
- f. Study the implementation of similar case algorithm - LTSFLinearForecaster in details from this [PR#4891](#).
- g. Implement the forecaster specific adapter only if needed.
- h. Implement the network class for architecture.
- i. Implement the forecasting class.
- j. Add test cases.

Midterm Evaluation (8 July - 12 July)

Phase - 2 (12 July - 19 Aug)

Week 7 (12 July - 15 July) + Week 8 (15 July - 22 July)

5. Implement the pytorch-forecasting adapter [Issue#4923](#)

- a. Study the [pytorch-forecasting](#) library in detail.
- b. Read [Dart Documentation](#) and see how its adapter is being carried out in [\[ENH\] darts adapter #5043](#).
- c. Create blue-print for adapter implementation.
- d. Create the base adapter class

- e. See and discuss if this adapter can utilise the implemented pytorch-adapter. For now, I think we cannot extend on the pytorch-adapter and need to implement a new one from scratch.

Week 9 (22 July - 29 July)

- 6. Implement DeepAR
 - a. Understand the [model usage](#) and its [architecture](#).
 - b. Implement DeepAR forecaster on top of pytorch-forecasting adapter
 - c. Add tests cases

Week 10 (29 July - 5 Aug)

- 7. Implement NBeats
 - a. Repeat the same steps as for Implementing NBeats.
 - b. Understand the [model usage](#) and its [architecture](#).
 - c. Implement DeepAR forecaster on top of pytorch-forecasting adapter.
 - d. Add tests cases

Week 11 (5 Aug - 12 Aug) + Week 12 (12 Aug - 19 Aug)

- 8. Huggingface Interface [#5790](#)
 - a. There is already work going on with implementing forecaster based on huggingface transformers [PR#5796](#)
 - b. Next step would be to add different fine-tuning methods like LoRA, LoHA, LoKr, AdaLoRA using PEFT library.
 - c. Although I have seen tutorials and use cases, yet never personally used any of the PEFT tuning methods. I would first spend some time understanding the library and its usage on time series transformer models available on huggingface.
 - d. Extend the huggingface interface under development with the

- PEFT methods during fit method.
- e. Extend the test cases for new interfaces.

Final Evaluation (19 Aug - 26 Aug)

Post GSOC (2 Sep -)

1. After having contributed this much to sktime, why would I stop there?
2. [What is my motivation?](#)

My Availability

- University Classes and Exams
 - I have my Final Semester Exams during [Week 1](#) of GSOC
 - I have a complete 3 month summer break from classes after that, during my GSOC coding period.
- Research Internship
 - I had applied for a research internship at [University of Kaiserslautern-Landau](#).
 - I'll have my flight scheduled during [Week 1](#) of GSOC.
- Management and Compensation
 - I have to take my GSOC [Week 1](#) completely off. Since I have my exams and flight to Germany.
 - I have created the work plan where I am compensating for this week's remainder work in other weeks.
 - I'll discuss thoroughly with the mentors during the community bonding period for the work hours and the project scope I can keep along with my research internship. I will make sure I make and approve a valid commitment that I can follow.

My Motivation

Ever since I came across github and the world of open-source, I wanted to become a part of it. The whole community and the process in these projects really gravitates me towards them.

I find myself very happy and lucky to find [sktime](#). The awesome community and extremely helpful mentors enabled me to start with my [contributions](#). I am impressed by the work culture of sktime and open-source as a whole and I want to become myself a driving force of this culture.

My long term plans are to keep up with sktime, and indulge in more open-source projects after completing GSOC. And perform my best to one day become a core developer at sktime!