Time Series Analysis and Forecasting

Faculty of DS & AI
Spring semester, 2026

Trong-Nghia Nguyen

Business AI Lab

Content

- About
- Group Requirements
- Environments
- Time-series concept with Python

About

Bio

[Website](#)

Trong-Nghia Nguyen (PhD)
(1996)

Present, Lecture at **National Economics University**

2021, Msc. at **Hanoi University of Science and Technology**

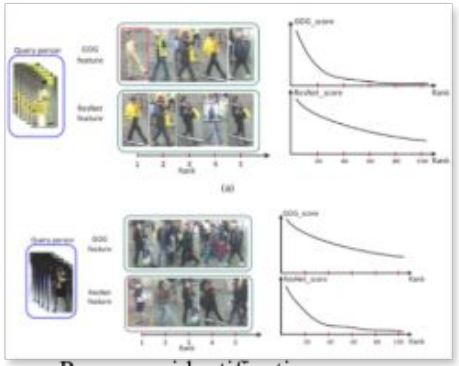
2025, PhD. at **Chonnam National University, South Korea**

2018, BS. at **University of Science, Hue University**

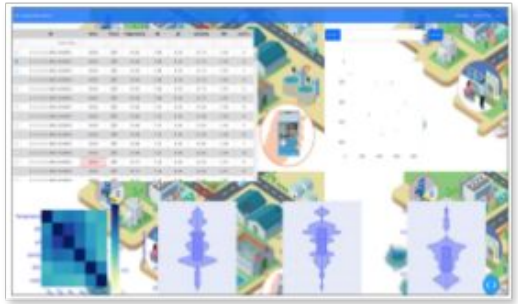
Disciplines: **Artificial Intelligence, Computer Science, Information Technology**

Research

Smart technology

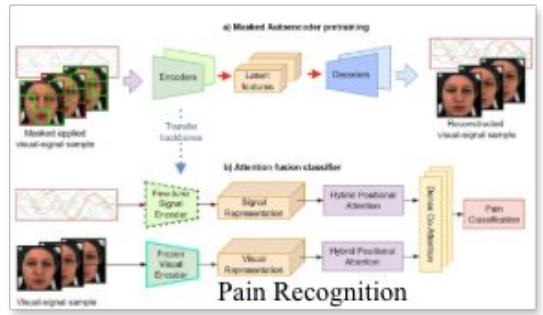


Person re-identification

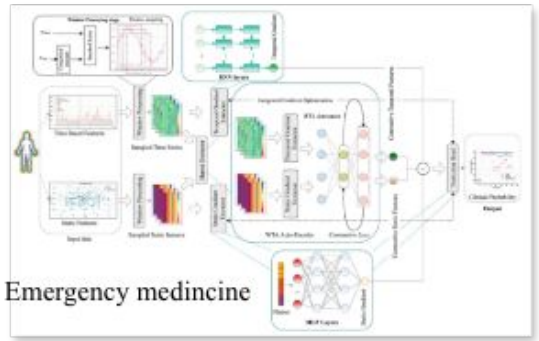


Smart Manufactoring

Smart Healthcare

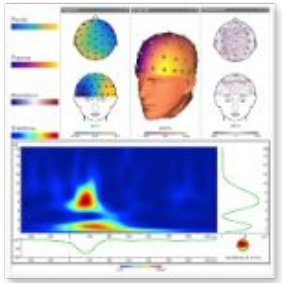


Pain Recognition

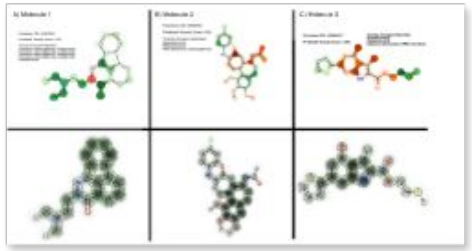


Emergency medicine

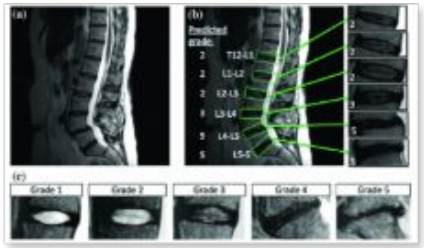
On going



Brain EEG signal



Drug's molecular structure



MRI - Pathology diagnostic

About

Course Overview

Course Website

- *Practical introduction to time series analysis & forecasting with Python*
- *Real-world applications in finance, economics, and healthcare*
- *Core statistical methods: ARIMA, SARIMA, exponential smoothing*
- *ML & DL extensions: tree-based models, LSTM, CNN*
- *Project-based learning with industry datasets*
- *See List of Topics*

About

Syllabus

Course Website

- Slides and reports could be made by LaTeX or .docx (or any) but should follow the template. **Incorrect template = Zero**.
- Every template for slides, reports, and poster could be found at Student_desk
- All textbooks are available online.

Attendance

10%
The team's member presentation

Midterm Baseline

30%
Report Poster + Presentation (Week 10)

Final Project

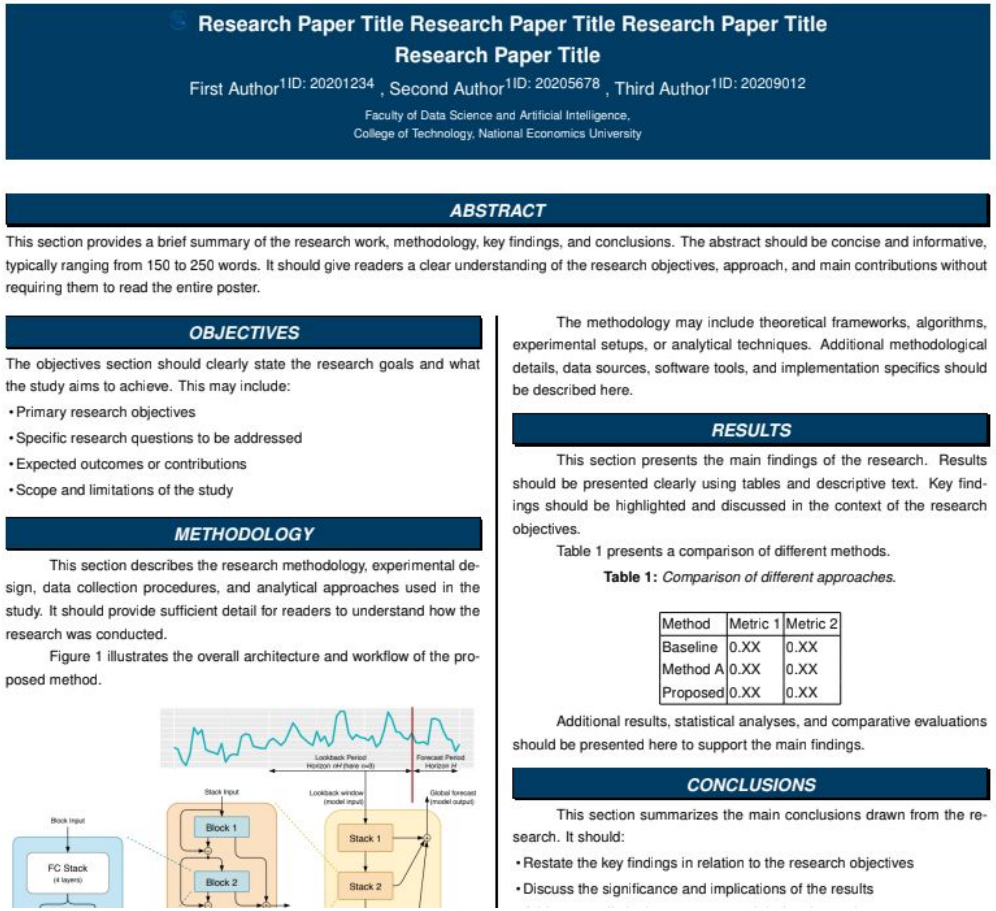
60%
Full project (paper) + presentation (Week 15)

About

Syllabus

Course Website

- All submissions except for source code must be submitted as a **.pdf file**.



Research Paper Title Research Paper Title Research Paper Title
Research Paper Title

First Author[1]ID: 20201234 , Second Author[1]ID: 20205678 , Third Author[1]ID: 20209012

Faculty of Data Science and Artificial Intelligence,
College of Technology, National Economics University

ABSTRACT

This section provides a brief summary of the research work, methodology, key findings, and conclusions. The abstract should be concise and informative, typically ranging from 150 to 250 words. It should give readers a clear understanding of the research objectives, approach, and main contributions without requiring them to read the entire poster.

OBJECTIVES

The objectives section should clearly state the research goals and what the study aims to achieve. This may include:

- Primary research objectives
- Specific research questions to be addressed
- Expected outcomes or contributions
- Scope and limitations of the study

METHODOLOGY

This section describes the research methodology, experimental design, data collection procedures, and analytical approaches used in the study. It should provide sufficient detail for readers to understand how the research was conducted.

Figure 1 illustrates the overall architecture and workflow of the proposed method.

The methodology may include theoretical frameworks, algorithms, experimental setups, or analytical techniques. Additional methodological details, data sources, software tools, and implementation specifics should be described here.

RESULTS

This section presents the main findings of the research. Results should be presented clearly using tables and descriptive text. Key findings should be highlighted and discussed in the context of the research objectives.

Table 1 presents a comparison of different methods.

Table 1: *Comparison of different approaches.*

Method	Metric 1	Metric 2
Baseline	0.XX	0.XX
Method A	0.XX	0.XX
Proposed	0.XX	0.XX

Additional results, statistical analyses, and comparative evaluations should be presented here to support the main findings.

CONCLUSIONS

This section summarizes the main conclusions drawn from the research. It should:

- Restate the key findings in relation to the research objectives
- Discuss the significance and implications of the results

Poster template

Group Requirements

Group working on weekly report

- Team formation: Groups of 1–4 students
- Dataset selection: 14 curated topics (finance, macro, health, climate, energy, etc.)
- All labs are project progress presentations.
- Midterm (Week 10): classical baseline (ARIMA/SARIMA)
- Final (Week 15): full project (classical + ML/DL)

Group Requirements

Group working on weekly report

- Weekly report: Just **update the slides** from last week; there's no need to create new slides.
- Midterm: **Presentation** & submit **poster**.
  - Midterm results should be a complete result of the baseline models.
- Final: Complete proposed method with **full report** & **complete slide**.

Group Requirements

Group working on weekly report

Topic Name

Progress Report

Group 1

Presenter: Tran A

Members: Le B, Nguyen Thi C, Dao Van D

Business AI Lab

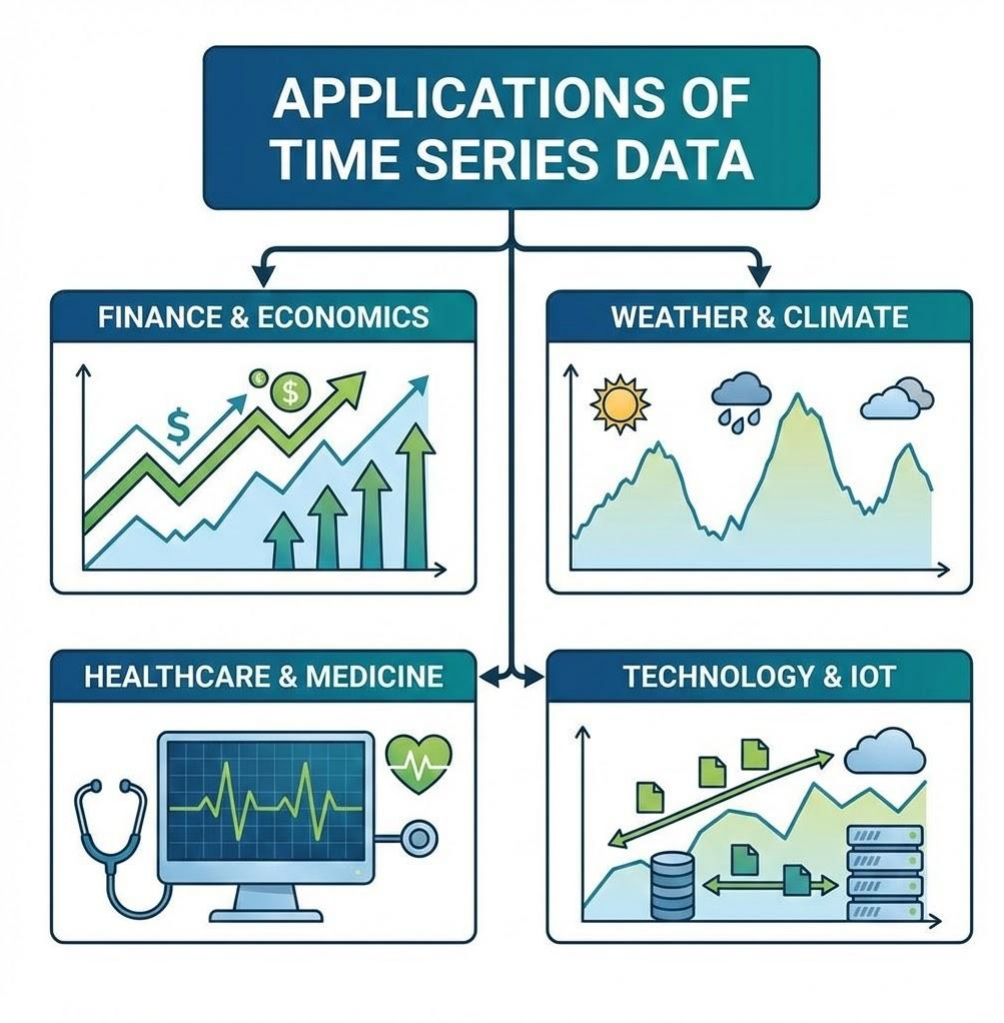
Slides template

Time-series concept with Python

What is time-series

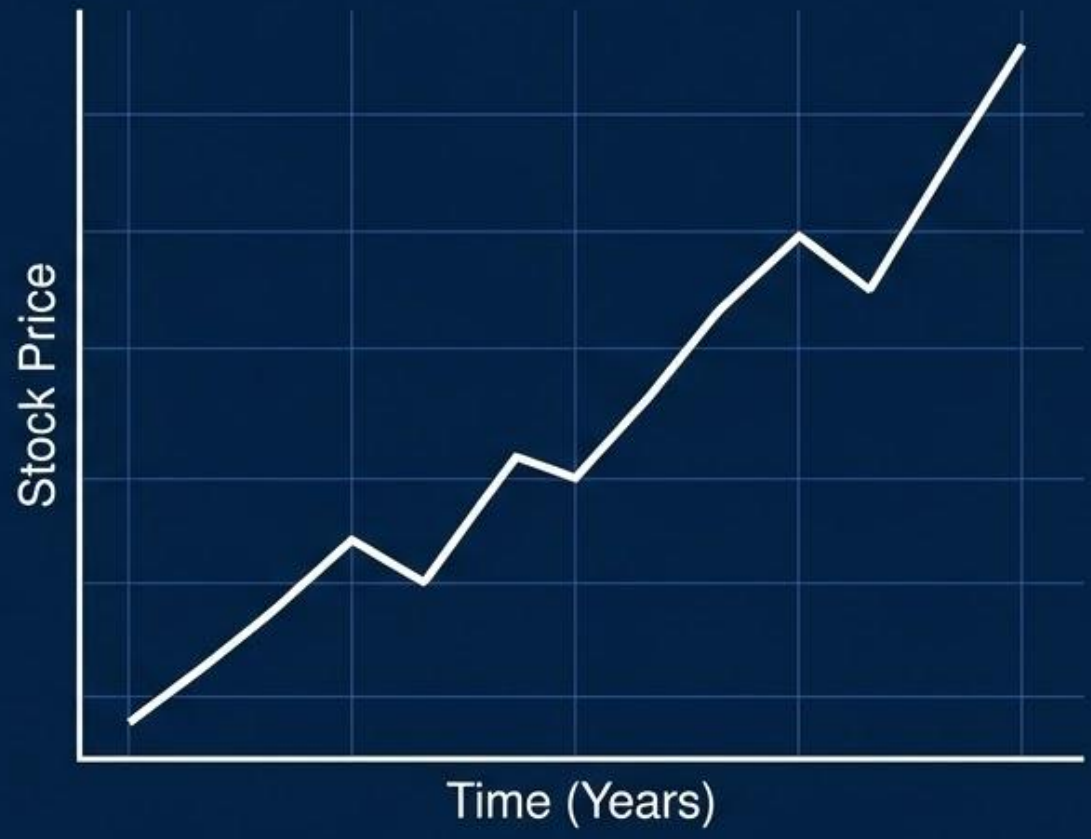
- Definition
  - A **sequence of observations** measured at successive **points in time.**
  - Ordered, temporal dependence between observations.

- Examples:
  - **Stock prices:** daily close price of AAPL or NVDA
  - **Economic indicators:** monthly unemployment, quarterly GDP
  - **Vital signs:** heart rate, blood pressure (ICU monitoring)
  - **Weather:** daily temperature, humidity (climate analysis)
  - **Energy:** hourly electricity demand, power consumption



Key Time Series Components - Trend

- **Trend**: The long-term movement of a time series, indicating a general direction (increasing, decreasing, or stable) over an extended period.
- Example: A company's stock price steadily increasing over several years.



Key Time Series Components - Seasonality

- **Seasonality**: Regular and repeating patterns within a fixed time period, such as a year, month, or week.
- Example: Retail sales consistently peaking in December each year.

Key Time Series Components - Cyclicality

- **Cyclicality**: Fluctuations or cycles that are not of a fixed period, often related to economic or business conditions, occurring over longer, irregular durations.
- Example: Multi-year business cycles of economic expansion and contraction.

Key Time Series Components - Irregularity (Noise)

- **Irregularity (Noise)**: Random, unpredictable variations in a time series that do not follow a pattern. They are often caused by unforeseen events or measurement errors.
- Example: Daily fluctuations in a sensor's reading due to random noise.

Time-series concept with Python

Python Ecosystem for Time Series

- **NumPy:** numerical computing, arrays, vectorization
- **Pandas:** data manipulation, time-based indexing, resampling
- **Matplotlib / Seaborn:** visualization and plotting
- **Statsmodels:** ARIMA, SARIMA, decomposition, tests
- **Scikit-learn:** ML models, preprocessing, pipelines
- **TensorFlow / PyTorch:** deep learning (LSTM, CNN)

Environments

Anaconda/MiniConda and Conda Environment Setup Instructions

- **Download and Install:**
  - **Anaconda:** Download the full installer from the Anaconda website. It includes Python, Conda, and a large suite of scientific packages.
  - **MiniConda:** Download the minimal installer from the Conda documentation page. It includes only Python and Conda, allowing you to install packages as needed.
  - Add conda to your PATH: conda init
- **Create a Conda Environment:**
  - Open your terminal or Anaconda Prompt.
  - Create a new environment with Python and necessary packages (e.g., `pandas`, `statsmodels` for time series) using the command:
    ```
    conda create --name timeseries_env python=3.10 pandas statsmodels jupyter
    ```
  - Activate the new environment:
    ```
    conda activate timeseries_env
    ```
- **Deactivate and Remove (Optional):**
  - Deactivate the environment: `conda deactivate`
  - Remove the environment: `conda remove --name timeseries_env --all`

Environments

Kaggle Setup and Dataset Loading

- **Register a Kaggle Account:**
  - Go to the Kaggle website.
  - Sign up using your email, a Google account, or a Facebook account.
  - Complete your profile setup.
- **Use Kaggle Notebooks:**
  - Navigate to the "Notebooks" section on Kaggle.
  - Click "New Notebook" to create a coding environment.
  - Select your preferred language (Python or R) and notebook type (Notebook or Script).
- **Load Pre-existing Kaggle Datasets:**
  - Find the dataset you want to use on the Kaggle Datasets page.
  - Open your new or existing Kaggle Notebook.
  - In the right-hand panel, click "+ Add Data" and search for the dataset.
  - Click the dataset to add it; the necessary input path will be automatically configured in your notebook.

Thank you!