Module 3: Part I — Foundations of Protein Language & AI Design (14th August - 20th August 2026)

Day 1·3Hours·Theory-firstwithguideddemo·Beginner-friendly

Overview

Format: 65% Theory · 25% Guided Demo · 10% Break

Prerequisites: Basic ML concepts, requires basic understanding of Python

Target: EGFR Kinase (cancer drug target)

Goal: Build conceptual clarity on protein language, PLMs, design strategies, and the AI tool landscape

Schedule

- 0:00 – 0:20 | How protein language works — alphabet, vocabulary, grammar
- 0:20 – 0:45 | Protein Language Models (PLMs) — ESM-2, masked residue prediction, embeddings
- 0:45 – 1:10 | How protein design works — forward vs inverse problem, design strategies
- 1:10 – 1:20 | Break
- 1:20 – 1:45 | ML architectures — Transformer · GNN · Diffusion Model
- 1:45 – 2:05 | Tool landscape — AlphaFold2/Chai-1/Boltz-1, ESM-2, ProteinMPNN, RFdiffusion, ColabFold
- 2:05 – 3:00 | Hands-on demo: Design a protein against EGFR kinase (55 min)

Key Concepts Covered

– The 20-letter amino acid alphabet — each residue is a token, each protein is a sentence
– Sequence → Structure → Function: the central mapping AI must learn
– Protein Language Models: ESM-2 learns by predicting masked residues — no labels needed
– Embeddings encode secondary structure, contacts, evolutionary signals, and functional sites
– Forward problem (AlphaFold2: sequence → structure) vs inverse problem (design: function → sequence)
– Evaluation Metrics for Protein Design pLDDT, iPAE scores.
– Design strategies: Fixed-backbone (ProteinMPNN), De novo generation (RFdiffusion), Hallucination (ESM-2 gradient)
– ML architectures: Transformer self-attention, GNN message passing, Diffusion denoising
– Tool map: what each model takes as input, produces as output, and when to use it

Hands-On Demo — Design a Protein Against EGFR (55 min)

- Step 01: Fetch EGFR kinase sequence from UniProt API using Biopython
- Step 02: Run ColabFold — predict 3D structure, inspect pLDDT confidence scores, visualise in NGLview
- Step 03: Run ProteinMPNN on predicted backbone — generate 20 candidate sequences
- Step 04: Score all 20 sequences with ESM-2 log-likelihood — rank top 5 by confidence
- Step 05: Re-predict top candidate structure with ColabFold — confirm RMSD < 1.5 Å vs original backbone

Tools Used

ColabFold, ProteinMPNN, ESM-2, UniProt API, NGLview, Biopython

Short Assignment:

Student will design a de novo protein using the AI models of their choice and also evaluate it

Module 1 Outcome

Students leave with a clear mental model of protein language, PLMs, and design strategies, along with a working notebook that produced their first AI-designed protein candidate against a real cancer target.

Module 3: Part II — Deep Learning Models: PLM Training, Sequence Design & Evaluation (21st August - 27th August 2026)

Day 2 · 3Hours · Hands-onlabs · Intermediate

Overview

Format: 15% Concept Primers · 75% Hands-on Labs · 10% Break

Prerequisite: Module 1 completed — students bring their notebook outputs

Target: Same EGFR kinase target, continued from Module 1

Goal: Train and Fine-tune a Protein Language Model for a specific Problem and Evaluate that model.

Schedule

- 0:00 – 0:10 | Primer: How LLMs works — Transformers architecture, understanding basics of Protein Language Modeling, Self-Attention, Concepts of Tokenization, Fine-tuning / Training a Transformer
- 0:10 – 1:15 | Lab 1: Run Colab notebook — Retrieval, Data Processing and Preparation
- 1:15 – 1:25 | Break
- 1:25 – 1:35 | Primer: Transformers architecture — how to choose correct architecture based on the problem
- 1:35 – 2:45 | Lab 2: Model Selection → Model Setup & Config → Model Training → Model Evaluation
- 2:45 – 3:00 | Debrief — review top candidates, connect outputs to Module 3 pipeline

Lab 1 — Data Retrieval & Preprocessing (65 min)

– Input: FASTA sequences retrieved from open source databases
– Step 1: Run Colab Notebook — retrieve protein sequences from Databases
– Step 2: Prepare Data — Data Filtering and Cleaning
– Step 3: Data Engineering — Identifying anomalies in data and perform data engineering
– Output: Dataset ready data split, feeding directly into Lab 2
– Fallback: Pre-processed data files if Colab runtime exceeds time limit

Lab 2 — Inverse Folding & Sequence Scoring (70 min)

– Input: Prepared Data from Lab 1
– Step 1: Model Selection → Understanding Model architecture and desired output
– Step 2: Setting up Model, Tokenization Technique & Model configuration
– Step 3: Model Training and Fine-tuning
– Step 4: Model Inference and Model Evaluation
– Validation metric: Perplexity, Log-Likelihood, Structure Prediction Metrics

Key Skills Developed

– Data Retrieval for PLMs — Retrieving Big Data for PLM models from Open databases
– Identifying and characterising protein sequences programmatically and applying Data Engineering
– Training Protein Language Models — understanding Model selection, configuration and training.
– Using temperature, fixed residues, and sampling, ESM-2 log-likelihood as a sequence quality filter
– Self-consistency validation: comparing designed vs. natural sequences

Tools Used

Colab, Python, ESM-2, pandas, CUDA, Torch, numpy, Transformers, LLMs

Module 2 Outcome

Each student leaves with a trained AI model, that can design protein sequences and validate by fold self-consistency.