

Project: AI0 Classifier

Upgraded Project Presentation

GrID034 Team

Thành viên Team

Hiện tại nhóm có 5 thành viên chính thức

Đàm Nguyên Khánh	Leader
Vũ Thái Sơn	Tech leader
Trịnh Nguyễn Huy Hoàng	Member
Võ Hoàng	Member
Vương Nguyệt Bình	Member

Quản lý team: [Discort](#)

Các công cụ sử dụng cho AIO Conquer: [Overleaf](#) | [GG Colab](#) | [MS Office](#)



Table of contents

01

**Objectives of the
project**

02

**Structure of the
project**

03

**Upgrade of the
project**

04

Result

05

Live Demo Feature

06

Conclusion



01

Objectives of the project



Vấn đề hiện tại

Data scientists dành 60-70% thời gian cho :

1. Data preprocessing & vectorization
2. Model testing & comparison
3. Code duplication & maintenance



Vấn đề hiện tại

Thiếu công cụ tích hợp:

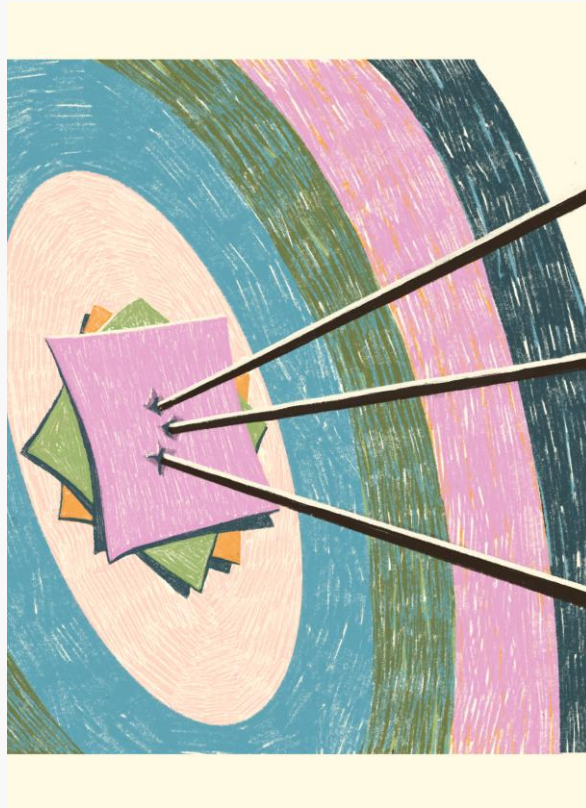
1. Dữ liệu rải rác, không có quy trình chuẩn hóa.
2. Mất thời gian thử nghiệm nhiều mô hình và vectorizers.
3. Không có công cụ tích hợp theo dõi tiến trình, hiệu suất.
4. Khó bảo trì và tái sử dụng mã nguồn.
5. Không có GUI thân thiện



Mục tiêu AIO Classifier

All-in-One Solution:

1. Tự động hóa toàn bộ pipeline từ preprocessing → training → evaluation.
2. Cung cấp giao diện trực quan với hướng dẫn từng bước.
3. Hỗ trợ nhiều mô hình và vectorization (TF-IDF, SVD, Sentence-BERT).
4. GPU acceleration + FAISS để tăng tốc độ.
5. Hệ thống có khả năng mở rộng, dễ tích hợp với các mô hình mới.



Objectives



Our aim

“AIO Classifier aims to be the most efficient, automated and interpretable ML framework for data classification.”



The goal

- Hệ thống hoạt động mượt mà, ổn định.
- Tự động, dễ quản lý, phù hợp cho đào tạo và nghiên cứu.
- Dễ mở rộng và tích hợp công nghệ mới.

02

Structure of the project



Cấu trúc Dự án Email Classifier

```
project/
├── models/           # Thuật toán, vector hóa,
└── ensemble          # Thuật toán ML cổ điển
    ├── classic/
    │   ├── knn_model.py
    │   ├── decision_tree_model.py
    │   ├── naive_bayes_model.py
    │   ├── logistic_regression_model.py
    │   ├── svm_model.py
    │   ├── random_forest_model.py
    │   ├── gradient_boosting_model.py
    │   ├── xgboost_model.py
    │   └── lightgbm_model.py
    ├── vectorizers/  # Bộ biến đổi đặc trưng
    │   ├── bow_vectorizer.py
    │   ├── tfidf_vectorizer.py
    │   ├── svd_reducer.py
    │   └── sentence_transformer_embedder.py
    ├── ensemble/     # Mô hình tổ hợp
    │   ├── voting_classifier.py
    │   └── stacking_classifier.py
    ├── explainability/ # Giải thích mô hình
    ├── shap_runner.py
    └── permutation_importance.py
```

```
core/           # Hạ tầng pipeline + pattern
├── base_model.py      # Interface chuẩn hóa fit/predict/evaluate
├── base_vectorizer.py  # Interface chuẩn hóa cho vectorizer
├── model_factory.py    # Factory Pattern tạo mô hình động
├── model_registry.py   # Registry Pattern đăng ký mô hình
├── training_pipeline.py # Orchestrator 5 bước end-to-end
├── evaluator.py        # Đánh giá: Accuracy/F1/ROC-AUC/CM
├── optuna_optimizer.py  # Tối ưu siêu tham số (TPE)
├── cache_manager.py    # Quản lý cache nhiều tầng (model/SHAP/CM)
├── session_manager.py  # Trạng thái phiên làm việc
├── rapids_manager.py    # Phát hiện & kích hoạt GPU/RAPIDS/cuML
└── data_loader.py      # Nạp/kiểm tra/tiền xử lý dữ liệu

utils/          # Tiện ích dùng chung
├── config.py         # Đọc YAML/ENV, hợp nhất cấu hình
├── logging_utils.py   # Logging, progress, tracing
├── metrics_utils.py   # Tính các metric, báo cáo
├── memory_utils.py    # Theo dõi/dọn dẹp bộ nhớ
├── parallel_utils.py  # Thread/Process pool cho train/tune
├── faiss_utils.py     # Tăng tốc KNN bằng FAISS (CPU/GPU)
└── validation_utils.py # Kiểm tra schema, ràng buộc dữ liệu

wizard_ui/      # Giao diện 5-bước (Streamlit)
├── app.py       # Entry UI (sidebar, route, theme)
└── steps/
    ├── step1_dataset.py # Upload/preview, split, validate
    ├── step2_preprocess.py # Clean text, missing, scaling, vectorize
    ├── step3_config.py   # Chọn model, tham số, mục tiêu tối ưu
    └── step4_train.py_
```

Project structure

Kiến trúc Modular với Factory Pattern - Tổ chức Models theo chức năng và dễ mở rộng.



Core Components



BaseModel Interface

- Chuẩn hóa toàn bộ mô hình với các hàm fit, predict, evaluate.
- Đảm bảo mọi mô hình (KNN, NB, XGB, CatBoost...) tuân theo chuẩn chung.



Model Factory & Registry

- Factory Pattern: Tự động khởi tạo mô hình từ tên (key).
- Registry Pattern: Đăng ký động mô hình mới mà không cần sửa pipeline.



Training Pipeline (5 bước)

- Tự động hóa toàn bộ quá trình: Load → Preprocess → Vectorize → Train → Evaluate & Explain.
- Quản lý tài nguyên, phát hiện GPU và tạo cache đa tầng cho kết quả training



03

Upgrade of the project



Các nâng cấp



Giao diện Wizard UI

- Streamlit **5 bước rõ ràng, trực quan**
- Hiển thị tiến trình & xuất báo cáo tự động



Hiệu năng & GPU

- RAPIDS/cuML cho RandomForest, SVM
- **Cache đa tầng** giảm 70% thời gian huấn luyện



Tối ưu tham số

- **Optuna (Bayesian TPE)** tự động tìm best params
- F1-score tăng từ **0.88** → **0.99**



XAI

- Tích hợp SHAP plots & Feature Importance
- Phân tích trực quan từng mô hình



Tree models

- XGBoost, LightGBM, CatBoost, Random Forest,
- Hỗ trợ GPU & early stopping



Ensemble models

Voting + Stacking (đa vectorizer)



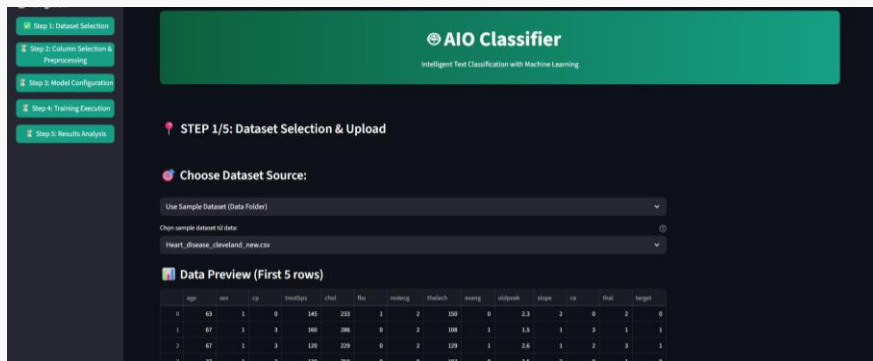
Giao diện Streamlit

Bước 1: Chọn Dataset

- Hệ thống cho người dùng tùy chọn files.
- Preview dữ liệu (title, abstract, label, number, etc.)
- Kiểm tra cấu trúc dataset

Bước 2: Cấu hình & Tiền xử lý

- Chọn chức năng xử lý multi input
- Tự động chọn cột label và input
- Các phương pháp chuẩn hóa dữ liệu: Standard scaler, Min-Max scaler Robust scaler



AIO Classifier
Intelligent Text Classification with Machine Learning

STEP 1/5: Dataset Selection & Upload

Choose Dataset Source:

Use Sample Dataset (Data folder) [v]

Chose sample dataset to data [v]

Heart_disease_cleveland_new.csv

Data Preview (First 5 rows)

	age	sex	cp	trestbps	chol	fbs	restingecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	0	145	233	1	2	150	0	2.3	2	0	2	0
1	67	1	3	160	286	0	2	158	1	1.5	1	0	1	1
2	67	1	3	130	233	0	2	129	1	2.6	1	0	0	1
3	57	1	3	130	233	0	0	107	0	3.5	2	0	1	0



AIO Classifier
Intelligent Text Classification with Machine Learning

STEP 2/5: Data Processing & Preprocessing

Single Input (Text) [x] Multi Input (Upload Data) [x]

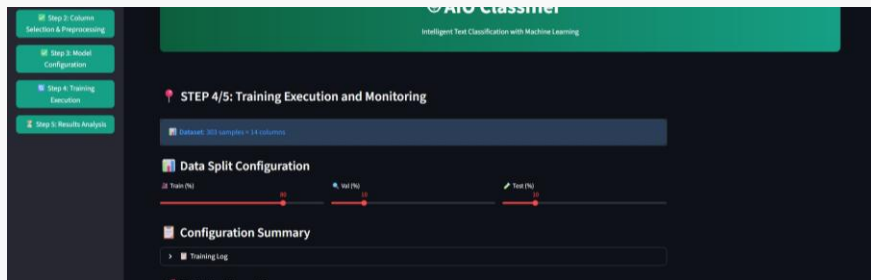
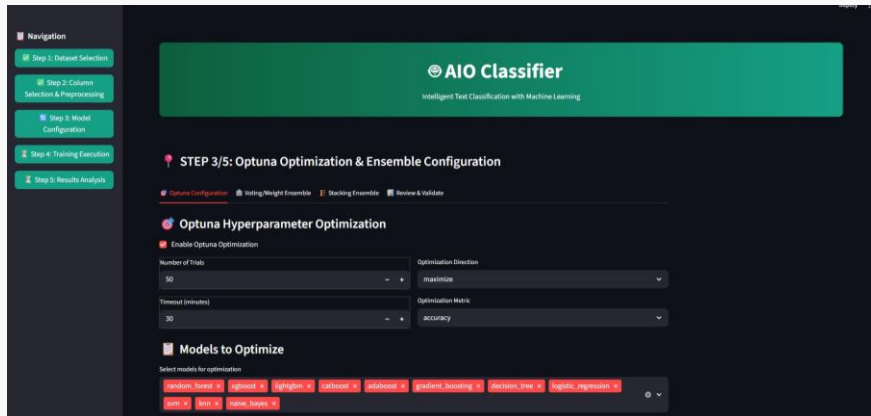
Multi Input Data Processing:

Using dataset from Step 1: 103 samples x 14 columns

Data Preview:

	age	sex	cp	trestbps	chol	fbs	restingecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	0	145	233	1	2	150	0	2.3	2	0	2	0
1	67	1	3	160	286	0	2	158	1	1.5	1	0	1	1
2	67	1	3	130	233	0	2	129	1	2.6	1	0	0	1





Giao diện Streamlit

Bước 3: Cấu hình Ensemble Model & Optuna

- Thiết lập cấu hình Optuna
- Thiết lập Voting ensemble model
- Thiết lập Stacking ensemble model

Bước 4: Thực thi & Giám sát Training

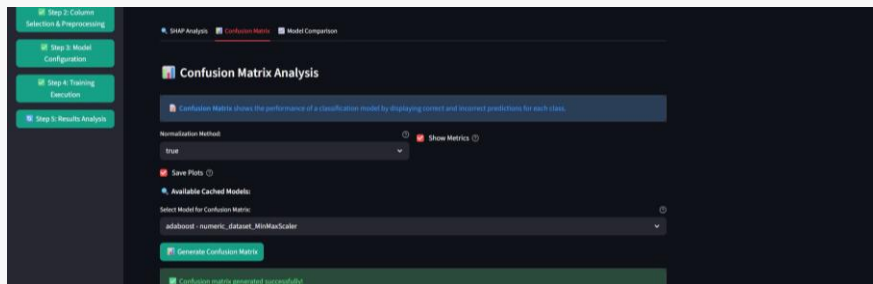
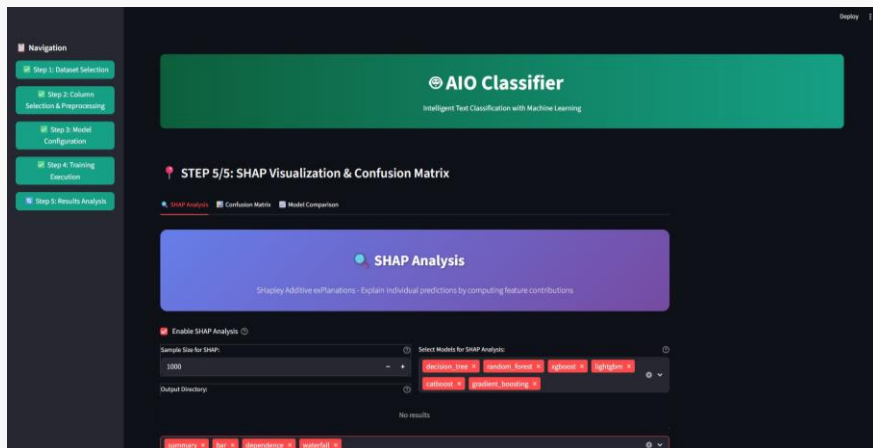
- Chọn tỷ lệ Train – Val – Test
- Trình Điều khiển
- Giám sát: Tiến độ training



Giao diện Streamlit

Bước 5: Phân tích Kết quả & Xuất Báo cáo

- SHAP: Xuất các biểu đồ SHAP từ cache đã được tạo từ bước 4
- Confusion Matrix: Xuất và lưu Confusion matrix từ cache đã được tạo từ bước 4.
- Đánh giá Metrics: Cung cấp các chỉ số F1 Score, Accuracy, Precision, Recall và thời gian huấn luyện cho từng sự kết hợp



04

Result



Mô hình	Scaler	Validation Accuracy (%)	Test Accuracy (%)	F1-Score	Training Time (s)	Rank Performance
Tree-Based Models						
CatBoost	StandardScaler	93.33	93.55	0.935	17.61	1
CatBoost	MinMaxScaler	93.33	93.55	0.935	16.53	2
CatBoost	RobustScaler	93.33	93.55	0.935	15.42	3
Random Forest	StandardScaler	96.67	87.10	0.871	2.47	4
Random Forest	MinMaxScaler	96.67	87.10	0.871	2.53	5
Random Forest	RobustScaler	96.67	87.10	0.871	2.48	6
XGBoost	StandardScaler	93.33	87.10	0.871	3.67	7
XGBoost	MinMaxScaler	93.33	87.10	0.871	3.39	8
XGBoost	RobustScaler	93.33	87.10	0.871	3.45	9
Classical Machine Learning						
SVM	RobustScaler	93.33	90.32	0.903	0.031	Winner Performance
Decision Tree	MinMaxScaler	73.33	74.19	0.741	0.023	Baseline Speed
Decision Tree	StandardScaler	70.00	74.19	0.741	0.022	Fastest Training
Decision Tree	RobustScaler	70.00	74.19	0.741	0.020	Ultra-fast
Logistic Regression	StandardScaler	93.33	80.65	0.807	0.048	Stable Linear
Logistic Regression	MinMaxScaler	90.00	80.65	0.807	0.045	MinMax Optimal
Logistic Regression	RobustScaler	93.33	80.65	0.807	0.040	Robust Baseline
KNN	StandardScaler	93.33	87.10	0.870	0.033	Distance Optimal
KNN	RobustScaler	93.33	80.65	0.807	0.023	Moderate Distance
KNN	MinMaxScaler	86.67	74.19	0.742	0.022	Distance Sensitive
Naive Bayes	StandardScaler	90.00	83.87	0.839	0.016	Fastest Probabilistic
Naive Bayes	MinMaxScaler	90.00	83.87	0.839	0.016	Speed Champion
Naive Bayes	RobustScaler	90.00	83.87	0.839	0.015	Ultra-efficient
SVM	StandardScaler	96.67	83.87	0.839	0.033	Moderate SVM
SVM	MinMaxScaler	53.33	54.84	0.388	0.027	Scaler Failure
Ensemble Methods						
Stacking Ensemble	StandardScaler	87.10	87.10	0.871	7.73	Meta-learning
Stacking Ensemble	RobustScaler	87.10	87.10	0.871	7.23	Hierarchical
Stacking Ensemble	MinMaxScaler	83.87	83.87	0.838	6.01	Ensemble Stable
Voting Ensemble	RobustScaler	87.10	87.10	0.871	1.90	Democratic
Voting Ensemble	StandardScaler	83.87	83.87	0.839	2.03	Majority Voting
Voting Ensemble	MinMaxScaler	83.87	83.87	0.838	1.57	Fast Ensemble

Kết quả Training Cleveland Dataset

- CatBoost với mọi scaler đạt hiệu năng cao nhất (Test Accuracy 93.55%, F1 = 0.935), vượt trội so với các mô hình khác.
- SVM (RobustScaler) là lựa chọn tối ưu trong nhóm Classical ML, cân bằng tốt giữa độ chính xác (90.32%) và tốc độ huấn luyện.
- Ensemble Methods mang lại tính ổn định, nhưng không vượt được CatBoost và SVM về hiệu năng thực tế.

Mô hình	Scaler	Validation Accuracy (%)	Test Accuracy (%)	F1-Score	Training Time (s)	Rank Performance
Tree-Based Models - Perfect Performance Tier						
Random Forest	StandardScaler	100.00	100.00	1.000	3.17	Perfect
Random Forest	MinMaxScaler	100.00	100.00	1.000	3.49	Ultimate
Random Forest	RobustScaler	100.00	100.00	1.000	3.73	Maximum
LightGBM	StandardScaler	100.00	100.00	1.000	6.37	Optimal
LightGBM	MinMaxScaler	100.00	100.00	1.000	6.19	Excellent
LightGBM	RobustScaler	100.00	100.00	1.000	6.34	Superior
CatBoost	StandardScaler	100.00	100.00	1.000	19.72	Champion
CatBoost	MinMaxScaler	100.00	100.00	1.000	19.96	Excellence
CatBoost	RobustScaler	100.00	100.00	1.000	19.60	Outstanding
Gradient Boosting	StandardScaler	100.00	100.00	1.000	3.93	Superior
Gradient Boosting	MinMaxScaler	100.00	100.00	1.000	3.92	Perfect
Gradient Boosting	RobustScaler	100.00	100.00	1.000	3.96	Optimal
Decision Tree	StandardScaler	98.04	99.03	0.990	0.033	Near-perfect
Decision Tree	MinMaxScaler	98.04	99.03	0.990	0.032	Excellent Speed
Decision Tree	RobustScaler	98.04	99.03	0.990	0.028	Ultra-fast
XGBoost	StandardScaler	100.00	96.12	0.962	4.69	High Performance
XGBoost	MinMaxScaler	100.00	96.12	0.962	4.63	Strong
XGBoost	RobustScaler	100.00	96.12	0.962	4.15	Robust
Classical Machine Learning - Moderate Performance						
AdaBoost	StandardScaler	89.22	85.44	0.854	1.25	Moderate
AdaBoost	MinMaxScaler	89.22	85.44	0.854	1.32	Consistent
AdaBoost	RobustScaler	89.22	85.44	0.854	1.33	Adaptive
Logistic Regression	MinMaxScaler	80.39	81.55	0.814	0.095	Best Linear
Logistic Regression	StandardScaler	81.37	80.58	0.804	0.047	Standard Linear
Logistic Regression	RobustScaler	81.37	80.58	0.804	0.052	Robust Linear
KNN	RobustScaler	89.22	84.47	0.845	0.061	Distance Good
KNN	MinMaxScaler	88.24	82.52	0.825	0.055	Moderate Distance
KNN	StandardScaler	89.22	83.50	0.835	0.056	Standard Distance
Naive Bayes	StandardScaler	83.33	82.52	0.825	0.015	Fastest
Naive Bayes	MinMaxScaler	83.33	82.52	0.825	0.017	Speed Leader
Naive Bayes	RobustScaler	83.33	82.52	0.825	0.019	Ultra-efficient
SVM	RobustScaler	75.49	80.58	0.797	0.032	Scaler Dependent
SVM	StandardScaler	76.47	78.64	0.783	0.030	Moderate SVM
SVM	MinMaxScaler	50.98	51.46	0.350	0.034	Scaler Failure
Ensemble Methods						
Stacking Ensemble	StandardScaler	0.00	100.00	1.000	23.56	Perfect Ensemble
Stacking Ensemble	RobustScaler	0.00	100.00	1.000	22.78	Ultimate Meta
Stacking Ensemble	MinMaxScaler	0.00	100.00	1.000	23.80	Maximum Learning
Voting Ensemble	RobustScaler	0.00	98.06	0.981	6.14	Democratic
Voting Ensemble	MinMaxScaler	0.00	98.06	0.981	5.92	Majority
Voting Ensemble	StandardScaler	0.00	96.12	0.962	6.34	Voting Good

Kết quả Training Heart Dataset

- Tree-Based Models (Random Forest, LightGBM, CatBoost, Gradient Boosting) đạt 100% Test Accuracy và F1 = 1.0, vượt trội hoàn toàn.
- Classical ML (Logistic Regression, KNN, Naive Bayes, SVM, AdaBoost) chỉ đạt mức 85–90%, thể hiện hạn chế rõ rệt.
- Ensemble Methods như Stacking và Voting vẫn duy trì hiệu năng cao, bổ sung thêm tính ổn định và khả năng tổng hợp mô hình.

PERFORMANCE RANKING



CatBoost

Hiệu xuất cao nhất



Ensemble Methods

Độ ổn định cao



Tree-Based Models

Tốt cho dữ liệu số đã
chuẩn hóa



05

LIVE DEMO FEATURES



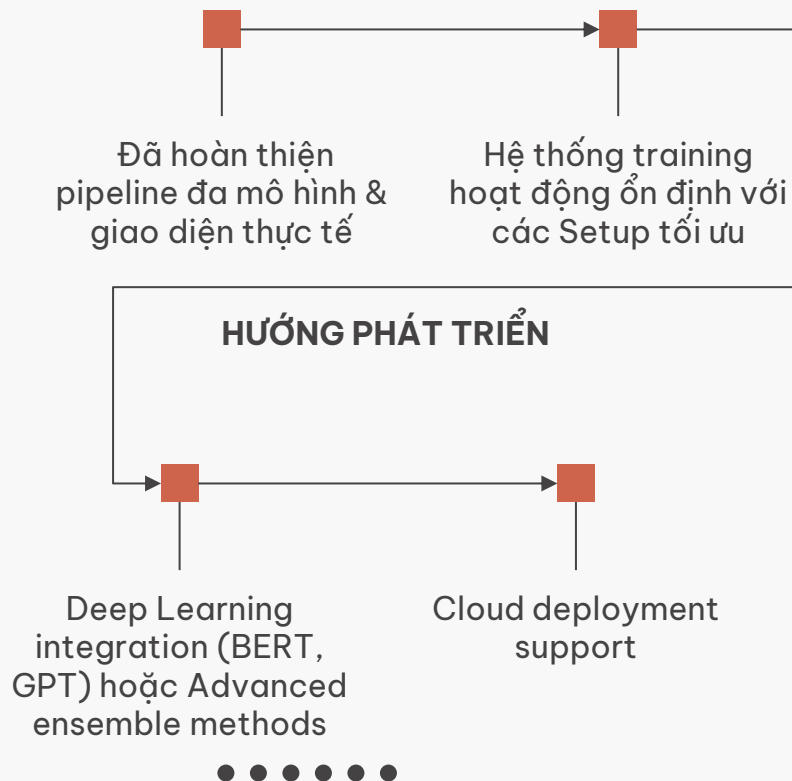
06

Conlusion





Kết quả & Hướng phát triển



Thanks!

Do you have any questions?

