**MẪU CẤU TRÚC BÀI BÁO Paper (TIẾNG VIỆT)**

Ghi chú: Các mục ghi “(tuỳ chọn)” có thể lược bỏ nếu không dùng. Điền nội dung vào các ngoặc <>.

**THÔNG TIN ĐẦU BÀI**

- Tiêu đề: <Tên đề tài, ví dụ: Dự đoán Khả năng Sống Sót trên Titanic bằng Học Máy và Khai thác Đặc trưng>

- Tác giả: <Họ và tên 1>, <Họ và tên 2>, ...

- Đơn vị: <Tên khoa/trường/công ty, địa chỉ>

- Email liên hệ: <email chính> (có cũng được ko có cũng được)

- Từ khoá: <Tối đa 5 từ khóa, ví dụ: Titanic, Feature Engineering, Logistic Regression, Tabular Data>

**ABSTRACT (TÓM TẮT)**

- Vấn đề: <nêu bối cảnh/nghiên cứu gì, tại sao quan trọng>

- Phương pháp: <pipeline/mô hình ngắn gọn>

- Kết quả chính: <con số tiêu biểu: Accuracy/F1/ROC-AUC/Kaggle score>

- Ý nghĩa: <đóng góp/thực tiễn/khả năng mở rộng>

**1. GIỚI THIỆU**

- Bối cảnh/Động lực: <vấn đề thực tiễn/bài toán benchmark>

- Thách thức: <thiếu dữ liệu, phân phối lệch, phi tuyến, v.v.>

- Mục tiêu: <xây mô hình gì, đánh giá ra sao>

- Đóng góp chính: <liệt kê 2–4 đóng góp cô đọng>

**2. CÔNG VIỆC LIÊN QUAN**

2.1. Mô hình cổ điển (ưu/nhược) → rút ra cách xử lý

- Nhóm mô hình: Logistic Regression, LDA, (khác)

- Ưu điểm: dễ diễn giải, ổn định khi chuẩn hoá/one-hot tốt

- Hạn chế: kém với quan hệ phi tuyến/tương tác phức tạp

- Hàm ý xử lý: cần Feature Engineering (tạo biến, binning, chuẩn hoá)

2.2. Mô hình nâng cao (ưu/nhược) → rút ra cách xử lý

- Nhóm mô hình: Random Forest, XGBoost/LightGBM/CatBoost, (Deep Tabular: TabNet, FT-Transformer)

- Ưu điểm: học tương tác/phi tuyến, hiệu năng tốt cho dữ liệu bảng

- Hạn chế: nhạy tham số, dễ overfit nếu dữ liệu nhỏ/tiền xử lý kém

- Hàm ý xử lý: chọn mô hình phù hợp quy mô dữ liệu; kiểm soát overfitting; giữ tính giải thích khi cần

**3. PHƯƠNG PHÁP ĐỀ XUẤT**

3.1. Tổng quan quy trình (data → EDA → model → train → evaluate → (deploy tuỳ chọn))

- Sơ đồ Pipeline: [Chèn hình: pipeline.png]

- Mô tả ngắn: <từ dữ liệu thô → xử lý MV → FE → chọn mô hình → huấn luyện → đánh giá>

3.2. Tổng quan vấn đề (Dataset Info)

- Nguồn dữ liệu: <Kaggle/URL/khác>

- Quy mô: <số mẫu, số thuộc tính>

- Biến mục tiêu: <mô tả nhãn>

- Các thuộc tính chính: <liệt kê nhanh: Pclass, Sex, Age, Fare, ...>

- Đặc điểm nổi bật: <cột thiếu nhiều, phân phối lệch, mất cân bằng lớp (nếu có)>

3.3. Mô hình đề xuất

- Mô hình gì: <Logistic Regression / XGBoost / (khác)>

- Lý do chọn: <ổn định, dễ diễn giải, hợp dữ liệu nhỏ; hoặc phi tuyến mạnh, SOTA tabular, v.v.>

- Thiết lập học & hiệu chỉnh tham số:

+ Cross-Validation: <k-fold, stratified?>

+ Siêu tham số: <liệt kê các hyperparameters chính, cách tìm: grid/random/Bayesian>

+ Tiêu chí chọn mô hình: <Accuracy, F1, ROC-AUC, …>

- Diễn giải mô hình (tuỳ chọn): <SHAP/LIME, hệ số LR, tầm quan trọng thuộc tính>

3.4. Cài đặt chi tiết (Implementation Details)

- Bước 1 – Xử lý Missing Values (MV): <Age/Fare→median; Embarked→mode; Cabin→drop/derive>

- Bước 2 – Khai thác đặc trưng (FE): <Title từ Name; FamilySize/IsAlone; Family\_Survival>

- Bước 3 – Rời rạc hoá (Binning): <Age bins; Fare bins (theo quantile)>

- Bước 4 – Chuẩn hoá dữ liệu: <StandardScaler cho biến số; one-hot cho biến phân loại>

- Bước 5 – Chọn lọc đặc trưng: <drop biến nhiễu/bias; giữ tập feature cuối>

- (Bước 6 – Tiền xử lý bổ sung) (tuỳ chọn): <xử lý ngoại lệ/outlier, cân bằng lớp, v.v.>

**4. THÍ NGHIỆM VÀ KẾT QUẢ**

4.1. Set up thí nghiệm

- Môi trường: Python <phiên bản>, scikit-learn <phiên bản>, (xgboost, pandas, numpy)

- Chia dữ liệu: <train/val/test hoặc CV; random\_state>

- Chỉ số đánh giá: <Accuracy, F1, ROC-AUC, …>

- Hình sử dụng (tuỳ chọn): missing values, bins, correlation, (SHAP)

4.2. Kết quả thu được

- (1) Bảng tóm tắt theo phiên bản (Version Comparison):

+ Bảng: Version | Thay đổi chính | Mô hình | Score

+ Hình (tuỳ chọn): version\_vs\_score.png

- (2) Bảng tóm tắt mô hình tối ưu (Model Comparison trên feature cuối):

+ Bảng: Model | Accuracy | F1 | ROC-AUC | Kaggle Score

+ (tuỳ chọn) Confusion Matrix, ROC, PR Curve

- (Thảo luận ngắn) (tuỳ chọn): <giải thích vì sao bước FE/mô hình A > B; hạn chế; độ ổn định>

**5. KẾT LUẬN**

- Tóm tắt đóng góp: <pipeline, FE quan trọng nhất, mô hình cuối>

- Kết quả chính: <con số tốt nhất>

- Hạn chế (tuỳ chọn): <dữ liệu nhỏ, thiếu biến,...>

- Hướng mở: <AutoFE, Bayesian Opt, Explainable AI, mở rộng dataset>

**TÀI LIỆU THAM KHẢO**

- [1] Kaggle Titanic dataset, URL

- [2] Pedregosa et al., Scikit-learn (JMLR 2011)

- [3] Chen & Guestrin, XGBoost (KDD 2016)

- [4] Lundberg & Lee, SHAP (NeurIPS 2017)

- [5] (Sách/Blog) Feature Engineering & Visualization

**PHỤ LỤC (tuỳ chọn)**

A. Cấu hình phần cứng/phần mềm chi tiết

B. Bảng siêu tham số (hyperparameters) cuối cùng

C. Mô tả bổ sung về tiền xử lý/các biến tạo thêm