

Pokemon Battle Predictor - Report

Alessandro Gautieri

gautieri.2041850@studenti.uniroma1.it

Tommaso Federici

federici.2214368@studenti.uniroma1.it

Lorenzo Colombini

colombini.1973692@studenti.uniroma1.it

1. Feature Engineering

1.1. Multi-Scale Temporal Windows

The battle log is segmented into three windows (w_1 : turns 1–10, w_2 : 11–20, w_3 : 21–30) to capture early/mid/late game dynamics. For each window, we track:

- **Damage dealt/taken:** offensive pressure per phase
- **KO and switches:** momentum and match control
- **Status afflictions:** strategic debuffs
- **Super-effective hits:** type matchup exploitation

This approach allows the model to distinguish aggressive early-game strategies from late-game comebacks.

1.2. Opponent's Unseen Type Prediction

Key innovation: we predict the types of unrevealed Pokémons using the global type distribution (from `predict.csv`). We calculate the **expected type advantage** by multiplying the probabilities of unseen types by our team's offensive multipliers. The feature `p1_expected_type_advantage_unseen_p2` captures the strategic potential against hidden Pokémons. This is possible knowing that the dataset is composed of real battles using the first 151 Pokémons, as battles in official tournaments have a typical distribution of types in teams.

1.3. Additional Key Features

- **Type Coverage Metrics:** super-effective/immune/resist counts per team, identify structural weaknesses
- **Offensive/Defensive Ratio:** $(Atk+SpA)/(HP+Def+SpD)$ measures team balance
- **HP Trajectory:** `avg_hp_pct_start`, `avg_hp_pct_end`, `delta` quantify trade efficiency
- **Interaction Features:** non-linear products between correlated features (e.g., damage \times speed, status \times HP_advantage) that GBDTs do not easily discover on their own

2. Training Strategy

2.1. Ensemble Architecture

Three complementary gradient boosting models:

- **LightGBM** (weight 0.431): fast, excellent on numerical features, handles class imbalance with `class_weight='balanced'`
- **CatBoost** (weight 0.284): robust on categorical features, reduces overfitting
- **XGBoost** (weight 0.284): strong L1/L2 regularization, predictive stability

2.2. Cross-Validation and Augmentation

- **10-fold Stratified CV:** preserves class distribution (50/50)
- **Side-Swap Augmentation:** doubles the training set by inverting the P1P2 perspective (mirror matchup with inverted label)
- **Seed Bagging** (3 seeds): averages predictions with different seeds to reduce variance
- **Isotonic Calibration:** per-fold and final, improves calibrated probabilities

2.3. Ensemble Method: Weighted Average

Grid search on OOF predictions identifies optimal weights (0.431, 0.284, 0.284). Superior to stacking as it is more robust and generalizes better on the test set.

3. Alternative Approaches

Linear models (LogisticRegression, SVM): underperformed (79–81% accuracy) due to inability to capture non-linear type interactions. **Stacking meta-learner:** marginally worse than weighted average (84.48% vs 84.71%), likely due to overfitting on OOF predictions.

Deep learning (TabNet): unstable training, no improvement over GBDTs. Final choice: weighted GBDT ensemble with Bayesian-optimized hyperparameters.