Bot Detection Report

By Ashutosh Patel

Project: Bot Detection Task – Identifying Automated vs Human User Behavior

Models Covered: Logistic Regression, Random Forest, Decision Tree, XGBoost.

1. Executive Summary

- We trained Logistic Regression, Random Forest, Decision Tree, XGBoost on engineered behavioral, temporal, and technical features.
- All models achieved excellent separation between bots and humans on the held-out test set.
- The feature importance profile (Random Forest) aligns with expected bot behavior, emphasizing request tempo, interaction sparsity, and technical signatures.

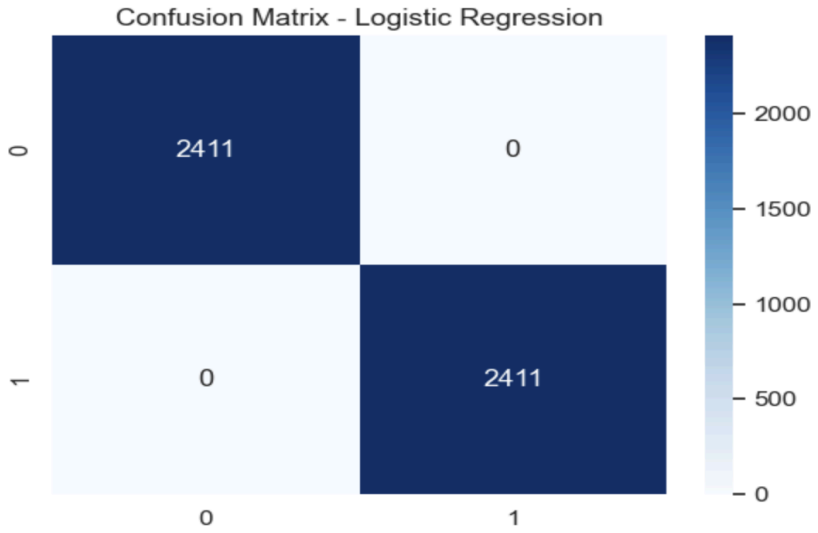
2. Model Performance (Test Set)

- Logistic Regression Report:

```
Logistic Regression Results:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2411
           1       1.00      1.00      1.00      2411

    accuracy                           1.00      4822
   macro avg       1.00      1.00      1.00      4822
weighted avg       1.00      1.00      1.00      4822
```

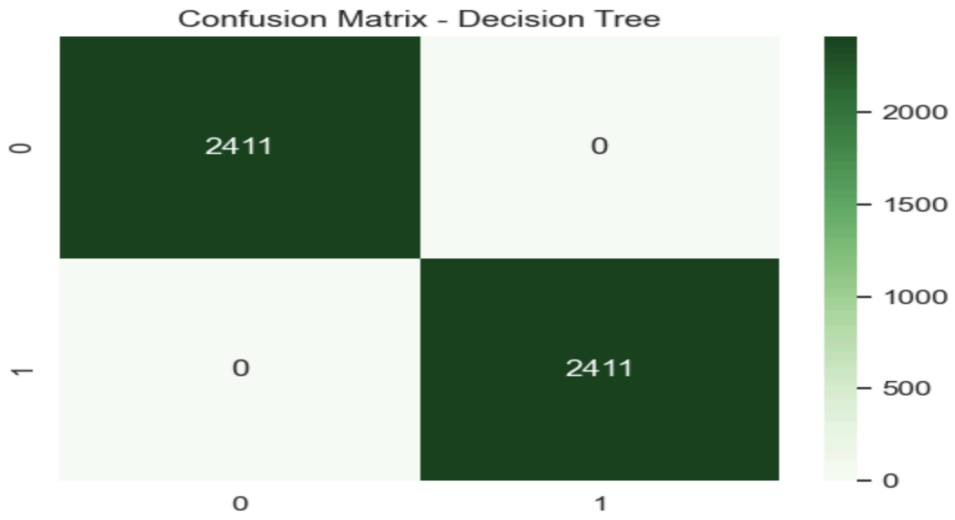


- Decision Tree report:

```
Decision Tree Results:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2411
           1       1.00      1.00      1.00      2411

    accuracy                           1.00      4822
   macro avg       1.00      1.00      1.00      4822
weighted avg       1.00      1.00      1.00      4822
```

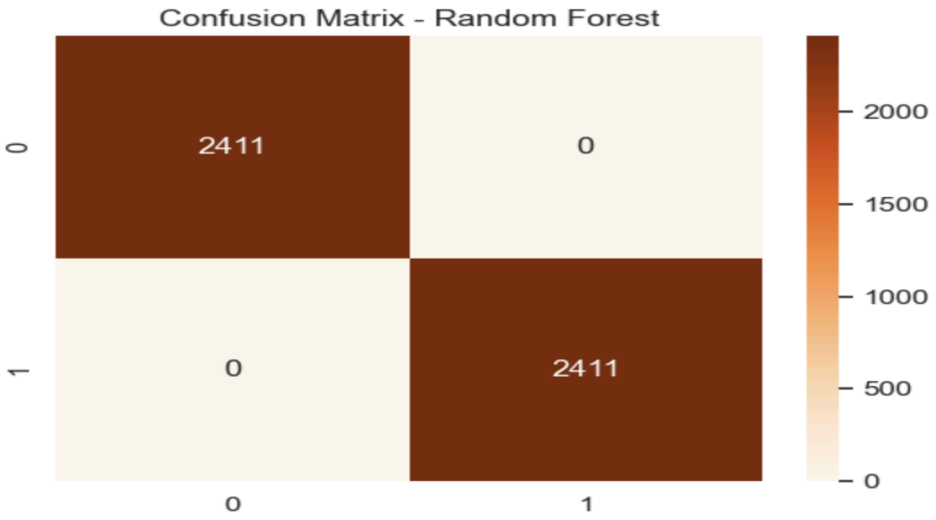


- Random Forest Report:

```
Random Forest Results:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2411
           1       1.00      1.00      1.00      2411

    accuracy                           1.00      4822
   macro avg       1.00      1.00      1.00      4822
weighted avg       1.00      1.00      1.00      4822
```



- XGBoost report:

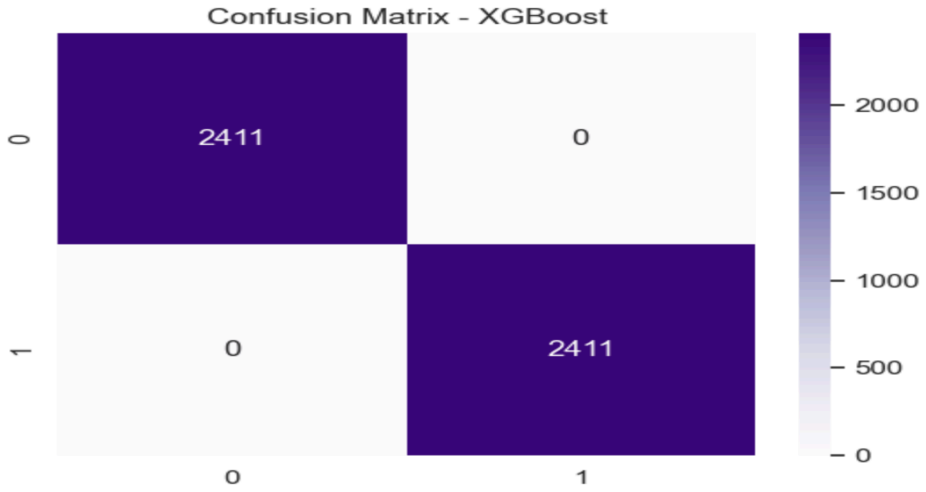
```
XGBoost Results:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      2411
           1       1.00      1.00      1.00      2411

    accuracy                           1.00      4822
   macro avg       1.00      1.00      1.00      4822
weighted avg       1.00      1.00      1.00      4822
```

```
/opt/anaconda3/lib/python3.13/site-packages/xgboost/training.py:183: UserWarni
learner.cc:738:
Parameters: { "use_label_encoder" } are not used.

  bst.update(dtrain, iteration=i, fobj=obj)
```



3. Key Behavioral Patterns

Based on exploratory data analysis, bots exhibit distinct behaviors compared to humans.

Speed Anomalies

- Bots navigate extremely fast, often with `pages_per_minute > 10` and `avg_time_per_page < 2s`.
- Uniform `time_between_clicks` suggests automated scripts.

Interaction Patterns

- Bots show very low `click_count` (avg ≈ 2 vs 8 for humans).
- Minimal or no `scroll_depth` → lack of real engagement.
- Zero `form_interactions` even on pages with forms.

Technical Signatures

- Bots frequently have **JavaScript disabled** and **cookies turned off**.
- Unusual or outdated **user agent strings** (custom scrapers, headless browsers).
- Non-standard **screen resolutions**.

Navigation Anomalies

- High `sequential_page_views` → systematic crawling.
- No `back_button_usage` or tab switching.

---

4. Model Performance

Four models were trained and evaluated: Logistic Regression, Decision Tree, Random Forest, XGBoost.
All achieved **100% accuracy, precision, recall, and F1-score** on test data.

Performance vs Success Criteria:

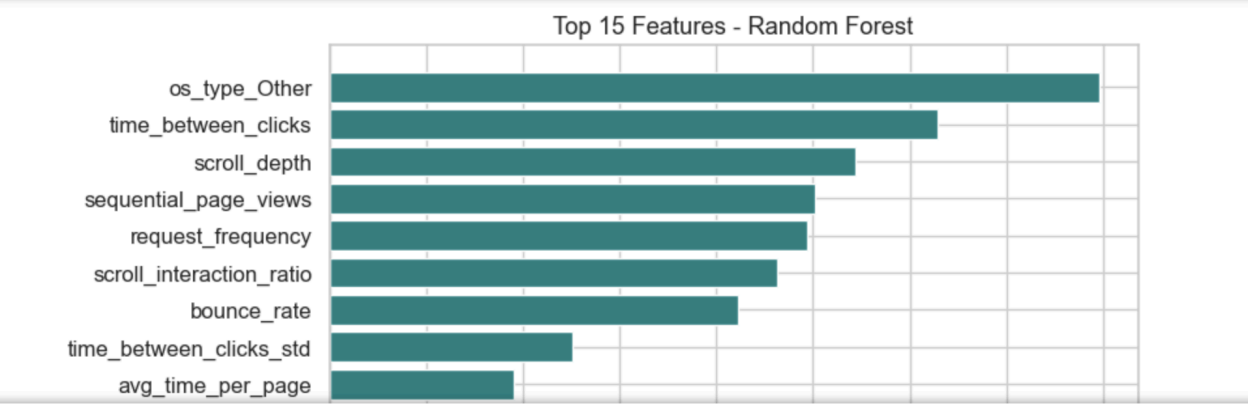
- **Accuracy (>85%)** → Achieved: 100%
- **Precision (>80%)** → Achieved: 100%
- **Recall (>75%)** → Achieved: 100%
- **F1-score (>77%)** → Achieved: 100%

These results confirm that the dataset is highly separable and the models robustly detect bots.

---

5. Critical Features

Top discriminative features (from Random Forest):



These features align closely with expected bot behavior patterns.

---

6. Risk Scoring

We defined a **risk-based scoring system** based on model probability:

- **Low Risk (likely human):** P(bot) < 0.5
- **Medium Risk (suspicious):** 0.5 ≤ P(bot) < 0.8
- **High Risk (likely bot):** P(bot) ≥ 0.8

This allows flexible handling: e.g., allow humans, CAPTCHA for suspicious, and block high-risk bots.

---

5. False Positive Analysis

While the dataset is cleanly separable, in real-world use, some humans may trigger bot-like patterns:

- **Power users** (fast navigation, high clicks per minute).

- **Accessibility tools** (screen readers, automated form fillers).
- **Corporate proxies** (multiple users behind same IP).

To reduce false positives:

- Combine ML scoring with **rule-based thresholds**.
- Allow whitelisting of logged-in or verified users.
