DADS6003

# Final project
# Paper R&D Practice

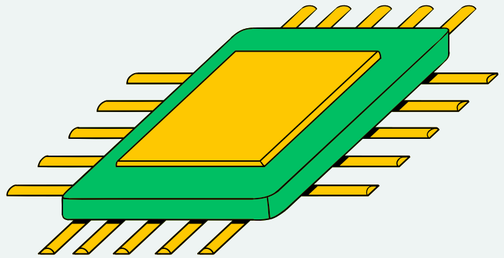
## Paper: "Evaluating spam filters and Stylometric Detection of AI-generated phishing emails"

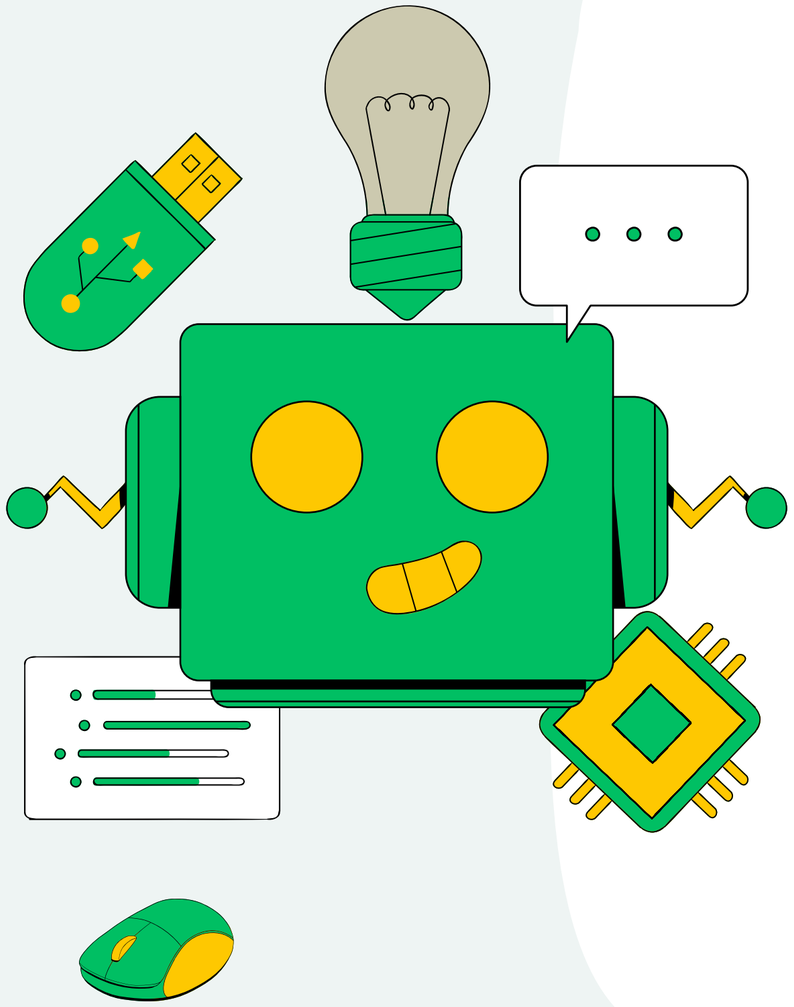
PRESENTED BY:


ณัฐรวิภาดา จิรวรัตน์ชัย 6720422004

จรรยพร ไหลกูล 6720422022

สโรชา เคนเวียง 6720422031

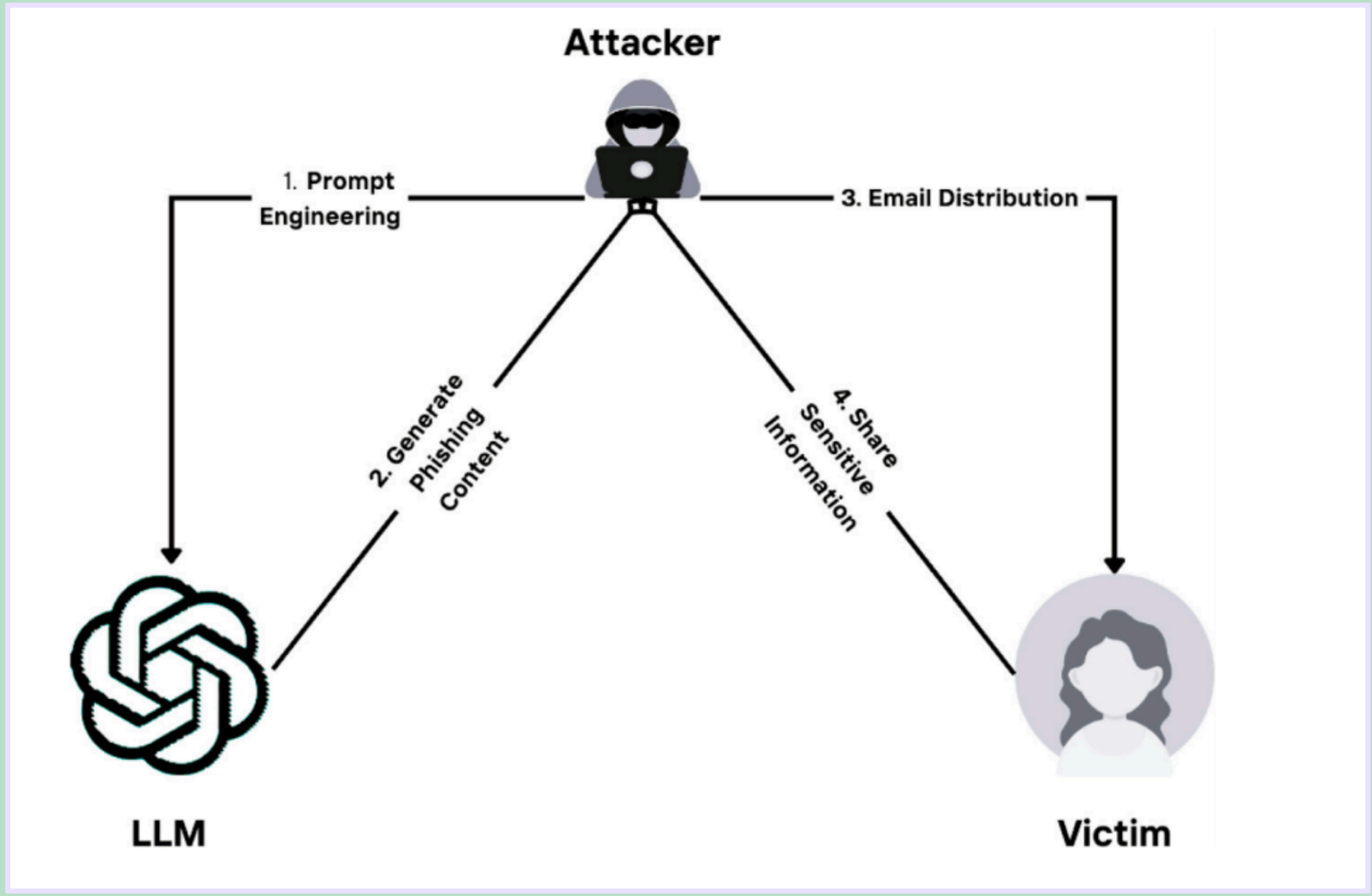





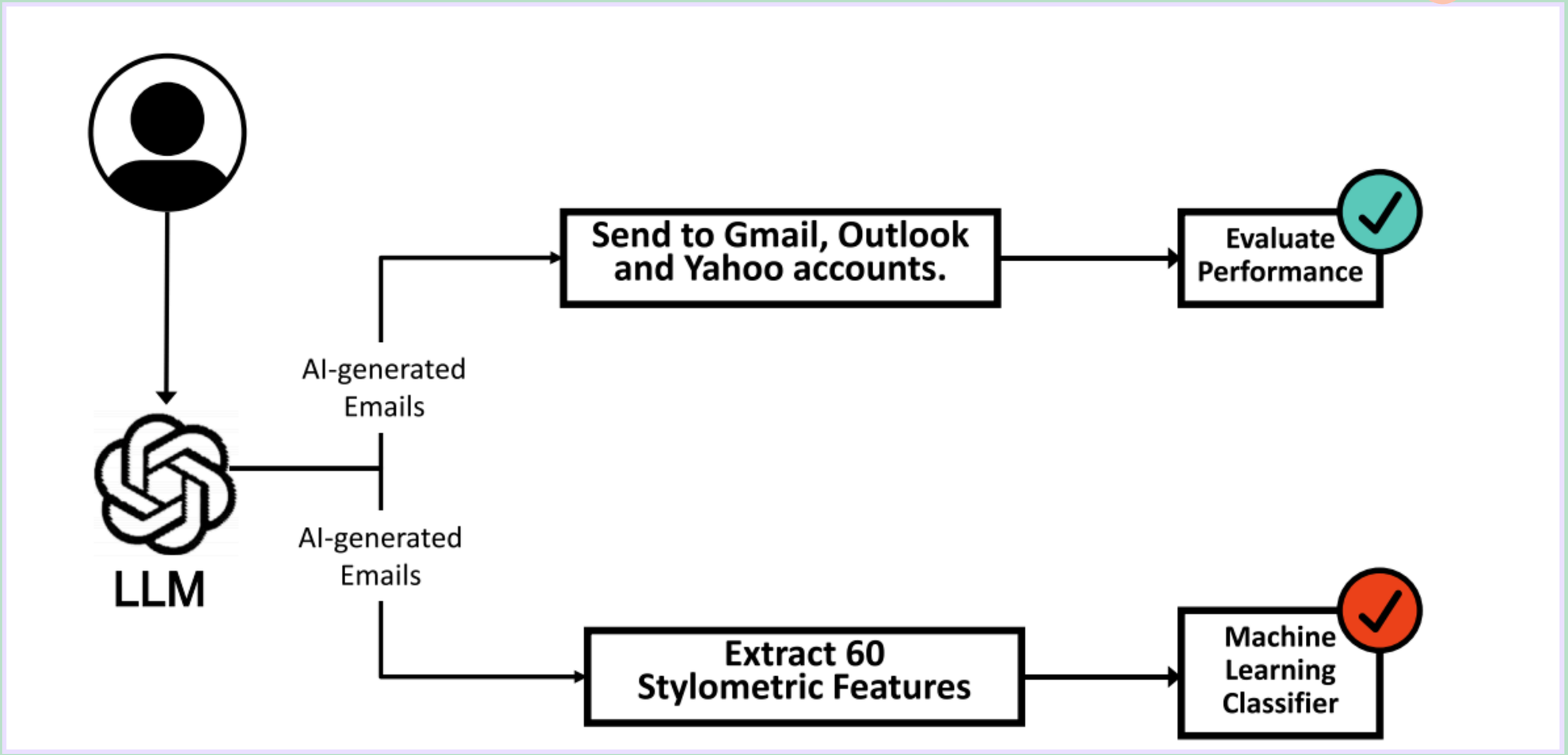
# AGENDA

- Introduction and Research Background :
- Pipeline Machine learning : Reproduce
  - Dataset Processing
  - Feature :
    - Stylometric Features from Paper
  - Modeling & Experiments (Reproduce)
    - Performance Comparison
- Additional Modeling  (New ideas)
  - More model and Features Added
  - Experimental Results
- Conclusion

# BACKGROUD

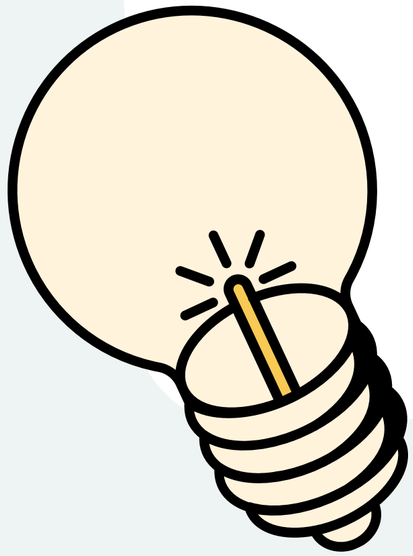

Attacker
1. Prompt Engineering
3. Email Distribution
2. Generate Phishing Content
4. Share Sensitive Information
LLM
Victim

# OVERVIEW OF THIS STUDY



# PIPELINE MACHINE LEARNING

1. **Data Preparation**
2. **Feature Preprocessing**
3. **Model Training**
   Modeling & Experiments
   (Reproduce)
4. **Model Evaluation**
5. **Deployment Or Analysis**



# DATA PREPARATION

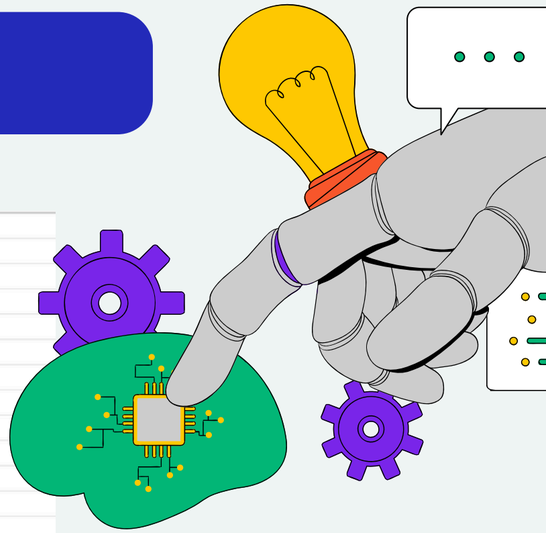
## 1.Dataset (Train–Test )

TRAIN.CSV

| Subject | Body | Label |
|---|---|---|
| An Exciting Opportunity to Engage in a Per | Dear David, | Legitimate |
| New Exciting Relationship Opportunity! Fr | Dear Julia, | Phishing |
| Important Account Notification Ã¢â‚¬â„¢ | Dear Toni, | Legitimate |
| Exclusive Financial Opportunity Awaits - Ir | Dear Chelsea, | Legitimate |
| Introducing Groundbreaking Health Enhan | Dear Christine, | Legitimate |
| Urgent Security Alert: Action Required for | From: TechZPro Security <no- | Phishing |
| Urgent Action Required: Strange Activities | Dear Robert, | Phishing |
| Security Alert & Account Verification Neec | Dear Melanie, | Phishing |
| Exciting Opportunity for a Personal Relatic | Hello Jonathan, | Legitimate |
| Exciting Job Opportunity with Flexible Wor | Dear Katherine, | Legitimate |
| IMPORTANT: Immediate Response Requir | Dear Alfred, | Phishing |
| Exciting Opportunity Awaits You - Verify Yc | Dear Kimberly, | Phishing |
| Exciting New Special Offer Just For You, M | Dear Marcus, | Legitimate |
| Important Technical Update - Action Requ | Dear Samantha, | Legitimate |
| Important Notice about Your MySecureBa | Dear Amanda, | Phishing |

TEST.CSV

| Subject | Body | Label |
|---|---|---|
| Exclusive Personal Relationship Oppo | Dear Sean, | Legitimate |
| Exceptional Work-From-Home Job Op | Dear Christopher, | Phishing |
| Notification of Recent Account Activi | Dear Kristen, | Legitimate |
| Exciting Work-from-Home Job Opport | Sender: HR Dept. | Legitimate |
| Exclusive Promotional Offer Just for Y | Dear Lindsay, | Legitimate |
| Exclusive Promotional Offer from "Dy | Dear Kyle, | Phishing |
| Exciting Job Opportunity Awaits You a | Dear Elizabeth, | Legitimate |
| Fantastic Work-From-Home Opportu | Dear John, | Phishing |
| Urgent Update Required for Your Hea | From: service@healtmngtsolution.us | Phishing |
| Exciting Job Opportunity and Immedia | [Acme Corp Logo] | Phishing |
| Immediate Job Opportunity at Tech S | Dear Mark, | Phishing |
| Important Technical Update - Securit | Dear Andrew, | Phishing |
| Important Account Notification - Plea | Dear Rebecca, | Legitimate |
| New Opportunity to Connect: Special | Dear Mike, | Legitimate |



## 2. Feature Normalization (StandardScaler)

```
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

## 3. Missing Value เติมค่าที่หายไปด้วยค่าเฉลี่ย

```
# เติมค่าที่หายไปด้วยค่าเฉลี่ย
X_train = X_train.fillna(X_train.mean())
X_test = X_test.fillna(X_test.mean())
```

## 4. การแปลง Label ของข้อมูลเป็นตัวเลข

```
# Encode target labels
label_encoder = LabelEncoder()
y_train_encoded = label_encoder.fit_transform(y_train)
y_test_encoded = label_encoder.transform(y_test)

class_names = label_encoder.classes_
print(f"คลาส: {class_names}")
```

## 5. รวมคอลัมภ์ Subject and Body เข้าด้วยกัน เป็น combined_text

```
# สร้างคอลัมน์ combined_text สำหรับการ extract features
if has_subject and has_body:
    train_df['combined_text'] = train_df['Subject'].astype(str) + " " + train_df['Body'].astype(str)
    test_df['combined_text'] = test_df['Subject'].astype(str) + " " + test_df['Body'].astype(str)
    print("✅ สร้าง combined_text จาก Subject และ Body")
else:
    text_column = train_df.columns[0]
    print(f"📝 ใช้คอลัมน์ '{text_column}' เป็นข้อความ")
    train_df['combined_text'] = train_df[text_column].astype(str)
    test_df['combined_text'] = test_df[text_column].astype(str)
```

# MODELING & EXPERIMENTS (REPRODUCE)

4 Models Used

1. Logistic Regression
2. Support Vector Machine (SVM)
3. Random Forest
4. XGBoost

Evaluation Metrics

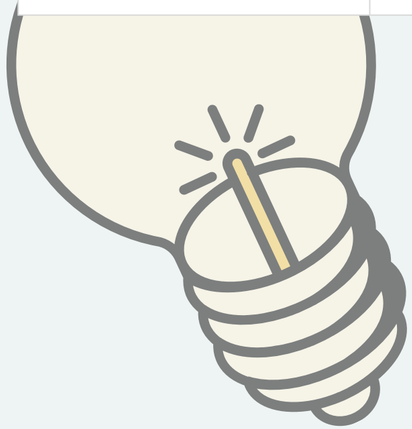
- Accuracy
- Precision
- Recall
- F1-Score
- AUC

Feature Importance + Ablation Study

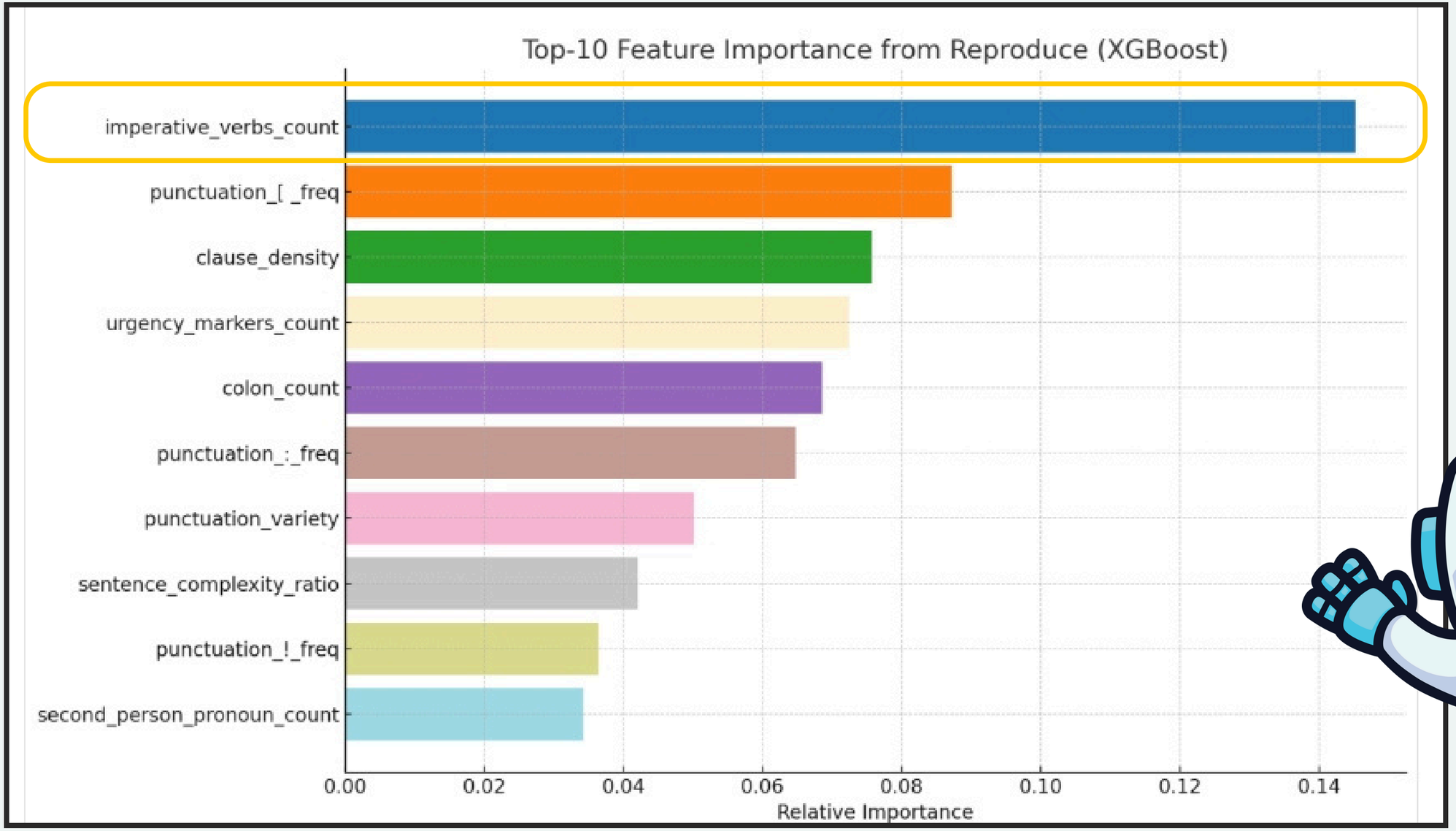
# MODELING & EXPERIMENTS (REPRODUCE) - RESULTS

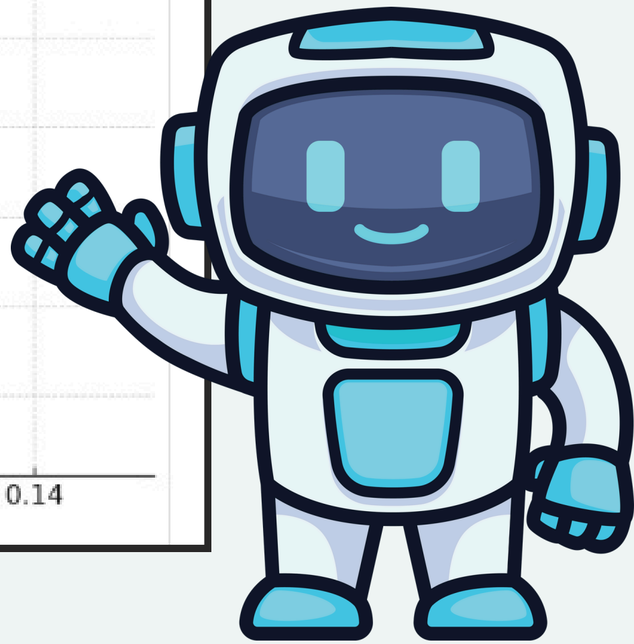
## Experimental Results : All of Model

| Algorithm | Accuracy | | | Precision | | | Recall | | | F1-Score | | | AUC-Score | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Modeling** | **Our** | **Paper** | **Diff** | **Our** | **Paper** | **Diff** | **Our** | **Paper** | **Diff** | **Our** | **Paper** | **Diff** | **Our** | **Paper** | **Diff** |
| **XGBoost** | 0.9231 | 0.96 | -0.04 | 0.9333 | 0.96 | -0.03 | 0.9231 | 0.96 | -0.04 | 0.9226 | 0.96 | -0.04 | 0.9704 | 0.99 | -0.02 |
| Logistic Regression | 0.6923 | 0.92 | -0.23 | 0.7124 | 0.93 | -0.22 | 0.6923 | 0.92 | -0.23 | 0.6848 | 0.92 | -0.24 | 0.8875 | 0.98 | -0.09 |
| Random Forest | 0.8846 | 0.92 | -0.04 | 0.8869 | 0.93 | -0.04 | 0.8846 | 0.92 | -0.04 | 0.8844 | 0.92 | -0.04 | 0.9704 | 0.99 | -0.02 |
| SVM | 0.8077 | 0.85 | -0.04 | 0.825 | 0.88 | -0.06 | 0.8077 | 0.85 | -0.04 | 0.8051 | 0.84 | -0.03 | 0.9586 | 0.96 | 0 |



# FEATURE IMPORTANCE (XGBOOST) REPRODUCE



# ABLATION STUDY

**ตรวจสอบฟีเจอร์กลุ่มใด “จำเป็นจริง ๆ” ต่อความแม่นยำของโมเดล**

## Ablation Study Results:

| Evaluation Metrics | Accuracy | Precision | Recall | F1-Score | Number Features |
|---|---|---|---|---|---|
| All Features | 0.9231 | 0.9333 | 0.9231 | 0.9226 | 61 |
| Without Top 10 | 0.8077 | 0.825 | 0.8077 | 0.8051 | 51 |
| Without Top 20 | 0.9231 | 0.9333 | 0.9231 | 0.9226 | 41 |
| Without Top 30 | 0.8462 | 0.8545 | 0.8462 | 0.8452 | 31 |

Top 10 คือ ฟีเจอร์กลุ่มนี้คือ กลุ่มหลัก ที่ช่วยแยก phishing vs legitimate
Top 30 features → Accuracy ลดลงอีกครั้ง ฟีเจอร์อันดับ 21–30 คือกลุ่ม “supportive signal ไม่ใช่ฟีเจอร์หลัก แต่ช่วยเสริมการแยกประเภท

# ADDITIONAL MODELING (NEW IDEAS)

**Assumption :**

**1.การเพิ่มฟีเจอร์ช่วยเพิ่มความแม่นยำในการทำนาย**

**2.นอกจาก 4 โมเดลที่กล่าวถึงแล้ว ยังมีโมเดลใดบ้างที่สามารถให้ความแม่นยำในการทำนายที่สูงกว่า?**

**7 Models Used**

1. L1 Regularization (SGD)
2. L2 Regularization (SGD)
3. Elastic Net (SGD)
4. K-Nearest Neighbors
5. Decision Tree
6. Neural Network
7. Gradient Boosting

**Evaluation Metrics**

- Accuracy
- Precision
- Recall
- F1-Score
- AUC

**+Additional Feature**

# ADDITIONAL MODELING (NEW IDEAS) - MORE MODEL

01

## L1 Regularization (Lasso)

- บังคับให้ค่าน้ำหนักบางตัวเป็น 0
- เลือกfeature selection อัตโนมัติ
- เหมาะกับข้อมูลที่มีฟีเจอร์จำนวนมาก
- โมเดลเรียบง่ายและตีความได้ดี
- อาจเกิด underfitting และไวต่อ noise

02

## L2 Regularization (Ridge)

- ไม่เซ็ตน้ำหนักเป็น 0 แต่ลดขนาดน้ำหนักทั้งหมดอย่างสม่ำเสมอ
- ช่วยเพิ่มเสถียรภาพของโมเดล และลด overfitting เสถียรและทั่วไปได้ดี
- ไม่สามารถตัดฟีเจอร์ที่ไม่สำคัญได้

03

## Elastic Net (SGD)

- ผสมระหว่าง L1 และ L2 เพื่อให้ได้ทั้งผลของ feature selection และความเสถียรของน้ำหนัก
- เหมาะเมื่อฟีเจอร์มีความสัมพันธ์กัน
- สมดุลระหว่าง L1 และ L2
- ต้องปรับ hyperparameter มากกว่าแบบเดี่ยว

04

## K-Nearest Neighbors(KNN)

- โดยทำนายจากตำแหน่งของจุดใหม่ แล้วดู k ตัวใกล้ที่สุด ของ training data
- เข้าใจง่าย,สามารถเรียนรู้ความสัมพันธ์แบบไม่เชิงเส้น
- ทำนายช้า,มีปัญหาเมื่อจำนวนฟีเจอร์สูง

05

## Decision Tree

- จำแนกข้อมูลด้วยการสร้างกฎแบบลำดับขั้นที่เลือกตัวแปรแบ่งกลุ่มเพื่อลดความไม่บริสุทธิ์ของข้อมูล
- ตีความง่ายและรองรับความสัมพันธ์ไม่เชิงเส้น
- เกิด Overfit หากไม่ควบคุมความลึก

06

## Neural Network

- โมเดลแบบหลายชั้นที่เรียนรู้ความสัมพันธ์ไม่เชิงเส้นผ่านการคำนวณของ neuron หลายชั้น
- เรียนรู้ pattern ซับซ้อนได้ดี
- ใช้เวลาเทรนมากและตีความยาก

07

## GradientBoosting

- วิธีรวมโมเดลอ่อนหลายตัวแบบลำดับ โดยให้แต่ละต้นไม้อธิบายส่วนที่โมเดลก่อนหน้าทำนายผิด
- ประสิทธิภาพสูงและลด bias ได้ดี
- ต้องปรับแต่งพารามิเตอร์และอาจใช้เวลาฝึกนาน

# ADDITIONAL MODELING (NEW IDEAS)


**01** **Emotional Tone Score - คะแนนโทนอารมณ์**

**positive_emotion_words**

**Example :** 'great', 'good', 'excellent', 'amazing', 'wonderful', 'happy', 'nice'

**negative_emotion_words**

**Example :** 'bad', 'terrible', 'awful', 'horrible', 'sad', 'angry', 'frustrated'

**02** **Formality Index - ดัชนีความเป็นทางการ**

**Example :** 'sincerely', 'regards', 'respectfully', 'cordially', 'yours'

**03** **Legal Terminology Density - ความหนาแน่นคำทางกฎหมาย**

**Example :**'terms', 'conditions', 'agreement', 'policy', 'compliance', 'regulation'

**04** **Sentence Structure Complexity - ความซับซ้อนโครงสร้างประโยค**

**คำนวณความแปรปรวนของความยาวประโยค**

**05** **Question Mark Density - ความหนาแน่นเครื่องหมายคำถาม**

**การใช้เครื่องหมายคำถามที่มากเกินไป ซึ่งอาจเป็นลักษณะการชักจูง**

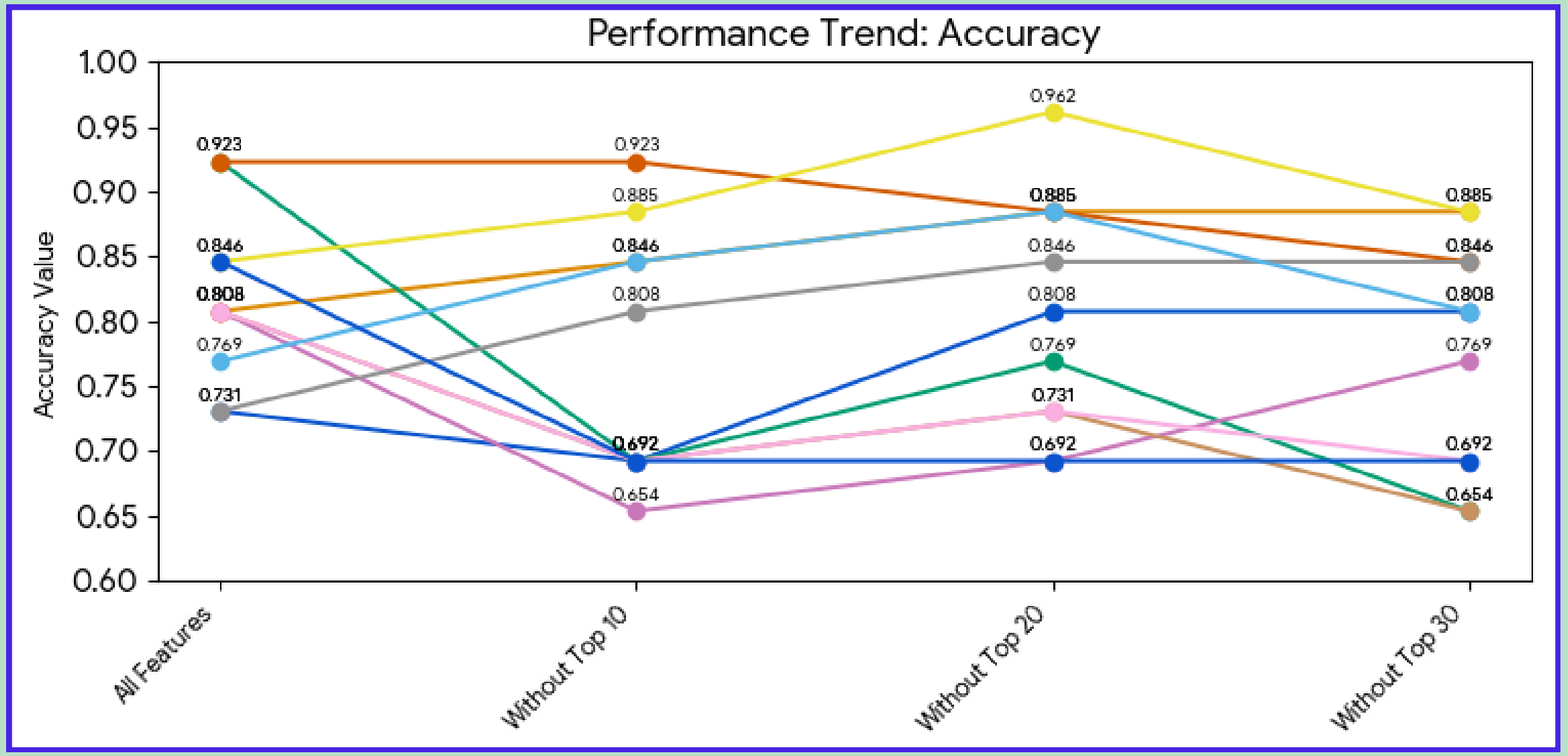
# Additional Modeling (New ideas)

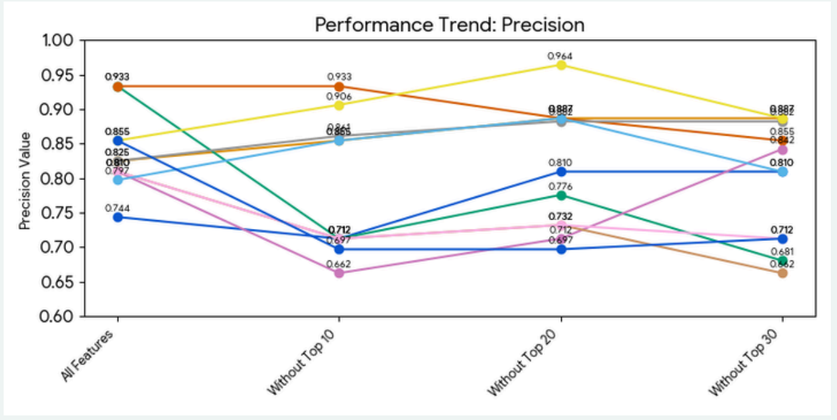
| Experimental Results : All of | | | | | |
|---|---|---|---|---|---|
| **Modeling** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **AUC-Score** |
| Random Forest | 0.9231 | 0.9333 | 0.9231 | 0.9226 | **0.9882** |
| XGBoost | 0.9231 | 0.9333 | 0.9231 | 0.9226 | **0.9704** |
| Decision Tree | 0.8462 | **0.8545** | 0.8462 | 0.8452 | **0.8462** |
| Gradient Boosting | 0.8462 | 0.8545 | 0.8462 | 0.8452 | **0.9586** |
| SVM | 0.8077 | 0.825 | 0.8077 | 0.8051 | **0.9586** |
| Elastic Net (SGD) | 0.8077 | 0.8095 | 0.8077 | 0.8074 | **0.8935** |
| L1 Regularization (SGD) | 0.8077 | 0.8095 | 0.8077 | 0.8074 | **0.9053** |
| L2 Regularization (SGD) | 0.8077 | 0.8095 | 0.8077 | 0.8074 | **0.9112** |
| Neural Network | 0.7692 | 0.7974 | 0.7692 | 0.7636 | **0.929** |
| Logistic Regression | 0.7308 | 0.7437 | 0.7308 | 0.7271 | **0.8994** |
| K-Nearest Neighbors (KNN) | 0.7308 | 0.825 | 0.7308 | 0.7271 | **0.8787** |

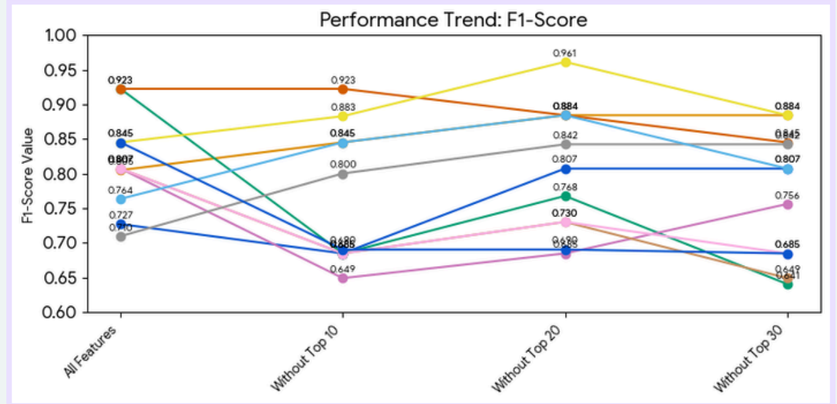
# ADDITIONAL MODELING (NEW IDEAS) -ABLATION STUDY

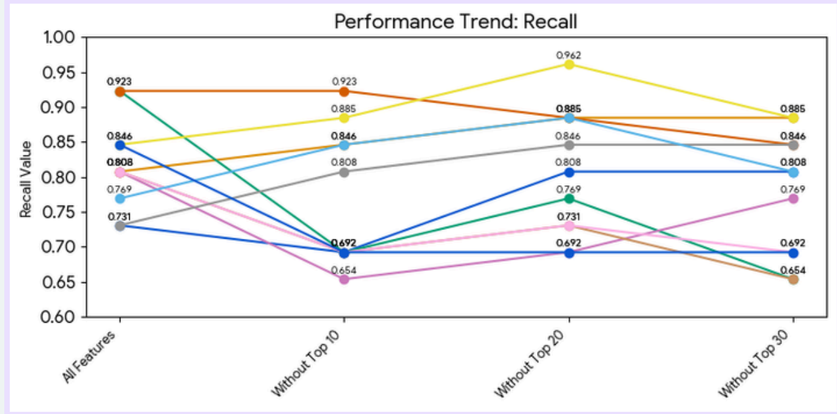
**Note**

Model
- Logistic Regression
- SVM
- Random Forest
- XGBoost
- L1 Regularization (SGD)
- L2 Regularization (SGD)
- Elastic Net (SGD)
- K-Nearest Neighbors
- Decision Tree
- Neural Network
- Gradient Boosting




- **XGBoost** เป็นโมเดลที่ค่าความแม่นยำไม่ได้เปลี่ยนแปลงมากเมื่อมีการลดฟีเจอร์ **แต่ Random Forest** ค่า Accuracy ลดลงอย่างมาก จาก 92% → 69% การลด feature ลงมีผลอย่างมาก

- ส่วนโมเดลอื่น เช่น Logistic Regression, Neural Network, KNN มีประสิทธิภาพลดลงเมื่อลดจำนวน Featue ลง เพราะ อาศัยจำนวนฟีเจอร์และความสมบูรณ์ของข้อมูลสูง

# Summary

## Reproduce

จาก Paper สรุปว่า XGBoost เป็น Model ที่ได้ค่า Accuracy เท่ากับ 96% แม่นยำที่สุด และจากการ Reproduce เราได้ค่า Accuracy เท่ากับ 92.3% เราสมมุติฐานจากการ Reproduce จากค่าที่แตกต่างกันว่า

- ขั้นตอนการทดลอง Preprocess ที่แตกต่างกัน
- ขั้นตอนการเขียน Coding แตกต่างกัน เช่น การ Tuning Hyperparameter ของแต่ละ Model ไม่เท่ากัน

## Additional Model

- ผลจากการเพิ่ม Sylometric Feature 5 ตัว และเปรียบเทียบ Accuracy ของแต่ละ Model พบว่าแต่ละ Model มีค่า Accuracy อยู่ในช่วง 73%-92.3%  โดยที่ค่า Random Forest และ XG Boost ได้ค่า Accuracy , Precision, Recall, F1-Score ที่เท่ากัน แต่แตกต่างกันที่ค่า AUC ซึ่ง **Random Forest ได้มากกว่าที่ 98.8%**

# THANK YOU