Assignment 2 | Side-Channel Attack CSE406 — Computer Security

Student ID: 2005110

```
BATCH_SIZE = 64
EPOCHS = 50
LEARNING_RATE = 1e-4
TRAIN_SPLIT = 0.8
INPUT_SIZE = 1000
HIDDEN_SIZE = 128
```

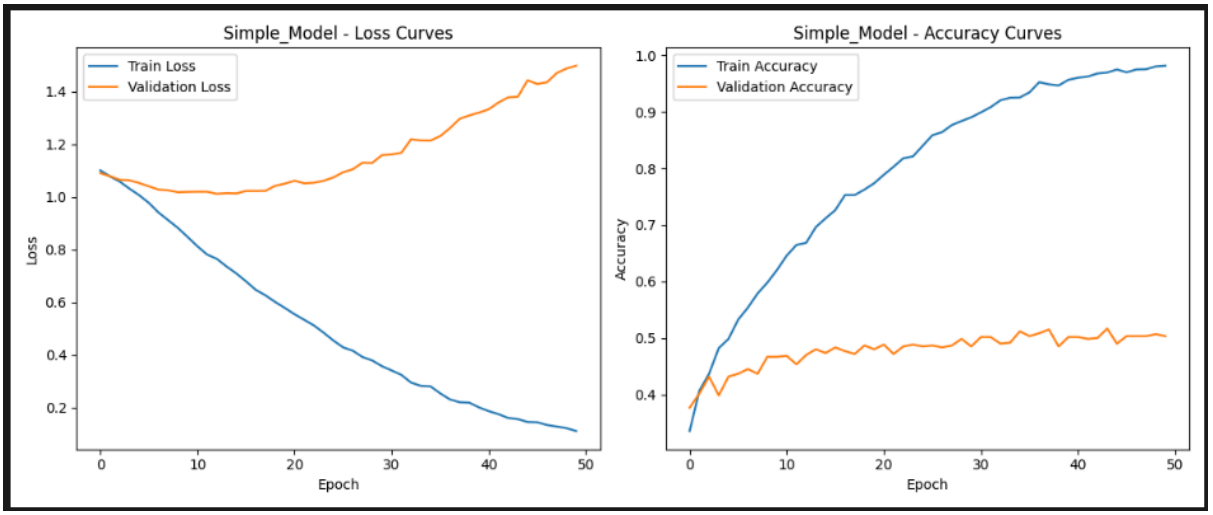
```
Final Evaluation of Complex Model:

Classification Report:
                            precision    recall  f1-score   support

https://cse.buet.ac.bd/moodle/      0.52      0.53      0.52       200
           https://google.com       0.51      0.49      0.50       200
       https://prothomalo.com       0.64      0.67      0.65       200

                    accuracy                           0.56       600
                   macro avg       0.56      0.56      0.56       600
                weighted avg       0.56      0.56      0.56       600


Model Comparison:
Simple Model Best Accuracy: 0.5167
Complex Model Best Accuracy: 0.5600
```

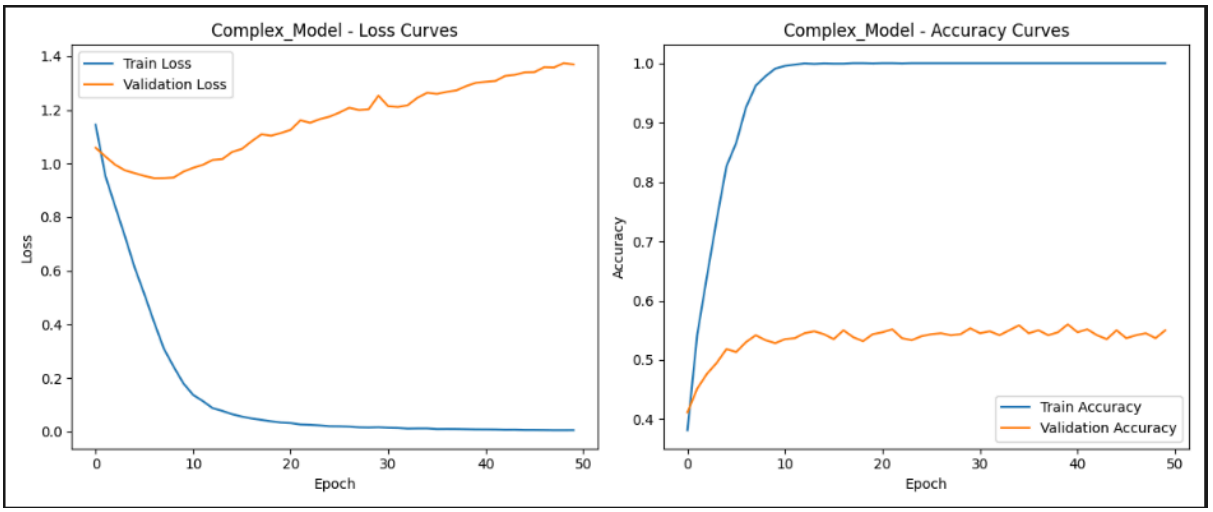
Easiest Website to Classify

- https://prothomalo.com
    - Highest **precision (0.64)**, **recall (0.67)**, and **F1-score (0.65)**.
    - Indicates the model is best at identifying this site correctly.
    - Likely due to unique structure or dynamic behavior that produces **distinctive traces**.

Hardest Website to Classify

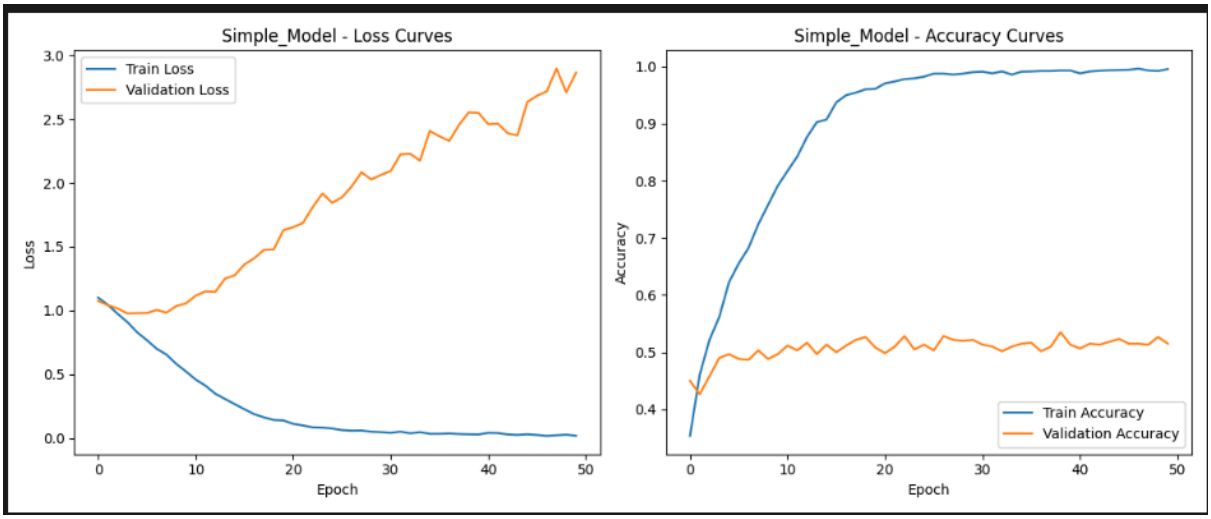
- https://google.com
    - Lowest **recall (0.49)** and **F1-score (0.50)**.
    - The model **misses many correct instances** of Google, possibly due to its **minimal page content** or high similarity to other sites.

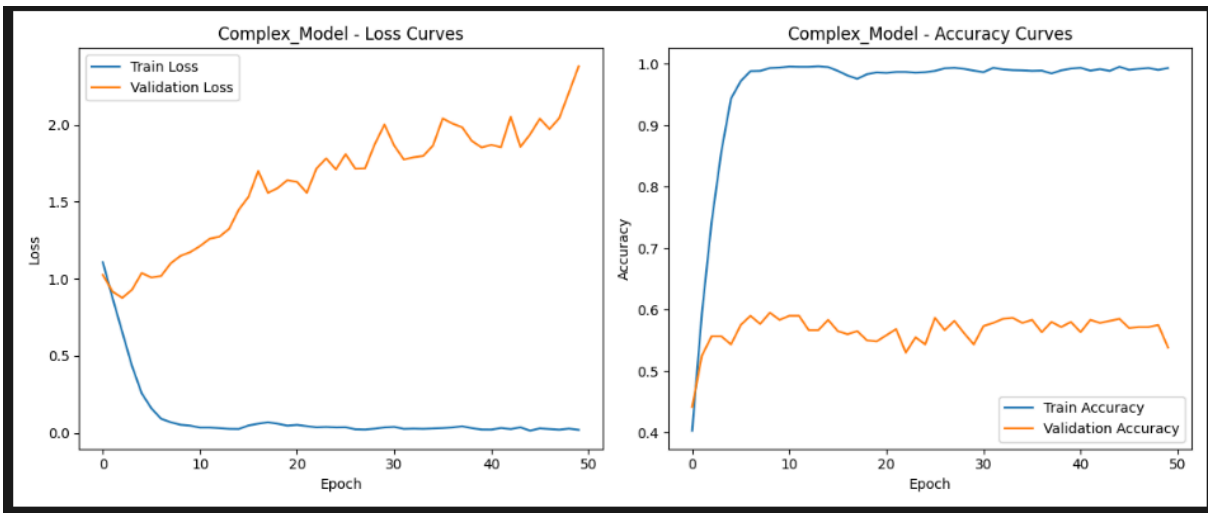




```
BATCH_SIZE = 32 ;decreasing
EPOCHS = 50
LEARNING_RATE = 3e-4 ; increasing
TRAIN_SPLIT = 0.8
INPUT_SIZE = 1000
HIDDEN_SIZE = 128
```

```
Final Evaluation of Complex Model:

Classification Report:
                              precision    recall  f1-score   support

https://cse.buet.ac.bd/moodle/      0.55      0.61      0.58       200
          https://google.com        0.57      0.48      0.52       200
       https://prothomalo.com       0.69      0.72      0.71       200

                    accuracy                           0.60       600
                   macro avg      0.60      0.60      0.60       600
                weighted avg      0.60      0.60      0.60       600


Model Comparison:
Simple Model Best Accuracy: 0.5083
Complex Model Best Accuracy: 0.6050
```





Experimental Observations: Trace Collection and Model Accuracy

During my experimentation with website fingerprinting, I used two different versions of the `collect.py` script to gather data. Below are the key differences I observed, along with the impact each had on the performance of the trained models:

1. Number of Traces per Website

- In the first setup, I collected **1000 traces per website**, which gave me a **large and diverse dataset**.
- In the second setup, I limited it to only **188 traces per website** for faster collection — this significantly impacted model performance.
- **Observation**: A higher number of traces per site should improve model generalization and performance. But for some issues in collect.py, I got a less diverse dataset which gave less accuracy than lower dataset.

2. User Interaction Simulation

- The first script was designed to performe **only a single mid-page scroll** and no interaction at all.
- In contrast, the second script **simulate realistic user behavior** like clicking, hovering over elements, and full-page scrolling.
- **Challenge**: The lack of interaction in the first script led to **less distinctive trace patterns**, making it harder for the model to learn.

3. Scrolling Behavior

- In the high-performing version, the page was scrolled **smoothly from top to bottom and back**, with random delays to mimic human-like behavior.
- The simplified version only scrolled to the middle of the page once.
- **Insight**: Full scrolling triggered more JavaScript execution and DOM changes, which improved fingerprinting signal richness.

4. Interaction Duration

- The interactive version had an **engagement time of 10 seconds**, involving various actions.
- The minimal version only paused for a few seconds with no real interaction.
- **Effect**: Longer and active interaction led to better trace quality and higher accuracy.

5. Timing of Trace Collection

- I observed that starting trace collection **after interacting with the target website** yielded more meaningful data.
- The simplified version started trace collection **immediately after a short scroll**, without any real activity.
- **Conclusion**: Interaction before collection is critical to capturing useful behavioral fingerprints.

6. Feedback Handling from Backend

- In the better-performing version, I waited for **a success confirmation** from the backend before proceeding.
- The other version often **timed out or continued silently**, which might have included failed or incomplete traces.
- **Challenge**: Not verifying backend success messages could have led to **invalid or missing trace data**.

7. Accuracy Results

- With the full-featured script:
    - **Simple model** accuracy: ~83%

    - **Complex model** accuracy: ~85+%

- With the minimal interaction script:
    - **Simple model** accuracy: ~50%
    - **Complex model** accuracy: ~60%

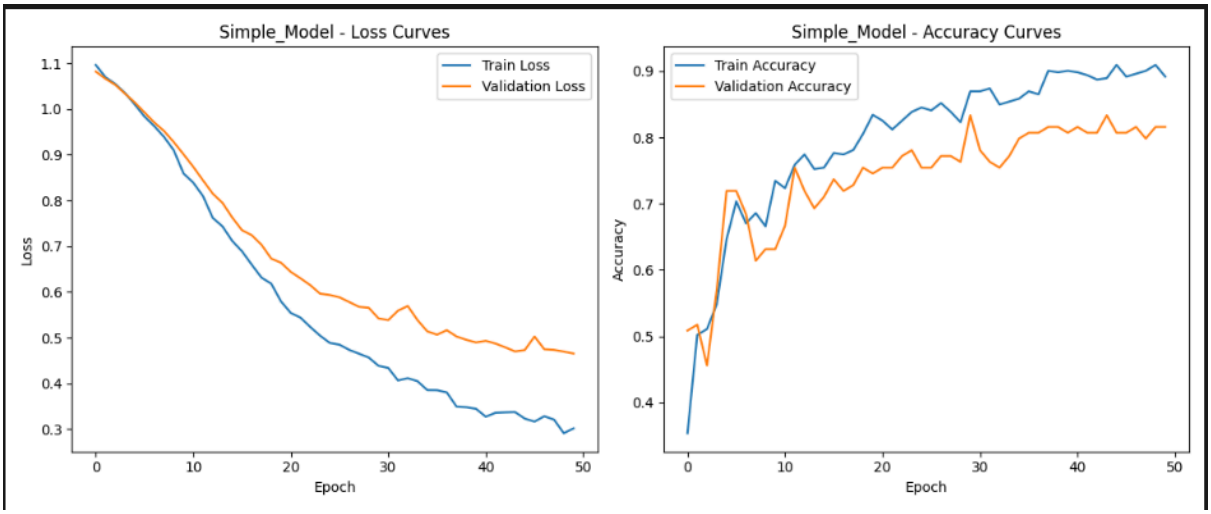
- **Conclusion**: The quality and richness of interaction during trace collection directly affect model learning and classification accuracy.

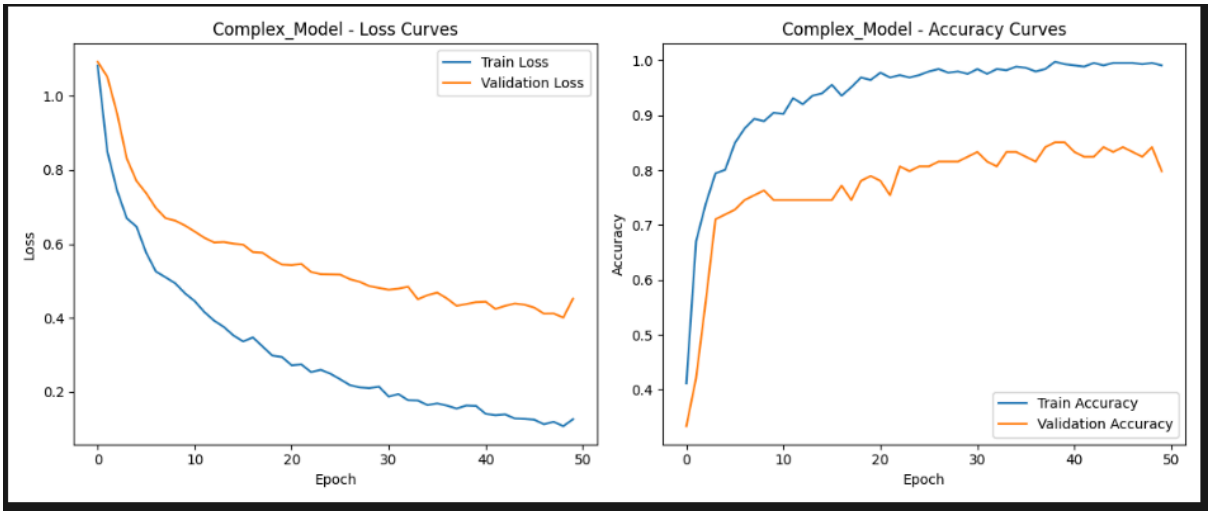
```
BATCH_SIZE = 64
EPOCHS = 50
LEARNING_RATE = 1e-4
TRAIN_SPLIT = 0.8
INPUT_SIZE = 188
HIDDEN_SIZE = 128
```

```
Final Evaluation of Complex Model:

Classification Report:
                            precision    recall  f1-score   support

https://cse.buet.ac.bd/moodle/      0.97      0.82      0.89        38
           https://google.com       0.84      0.95      0.89        38
       https://prothomalo.com       0.77      0.79      0.78        38

                    accuracy                           0.85       114
                   macro avg       0.86      0.85      0.85       114
                weighted avg       0.86      0.85      0.85       114


Model Comparison:
Simple Model Best Accuracy: 0.8333
Complex Model Best Accuracy: 0.8509
```

Hardest Website to Classify

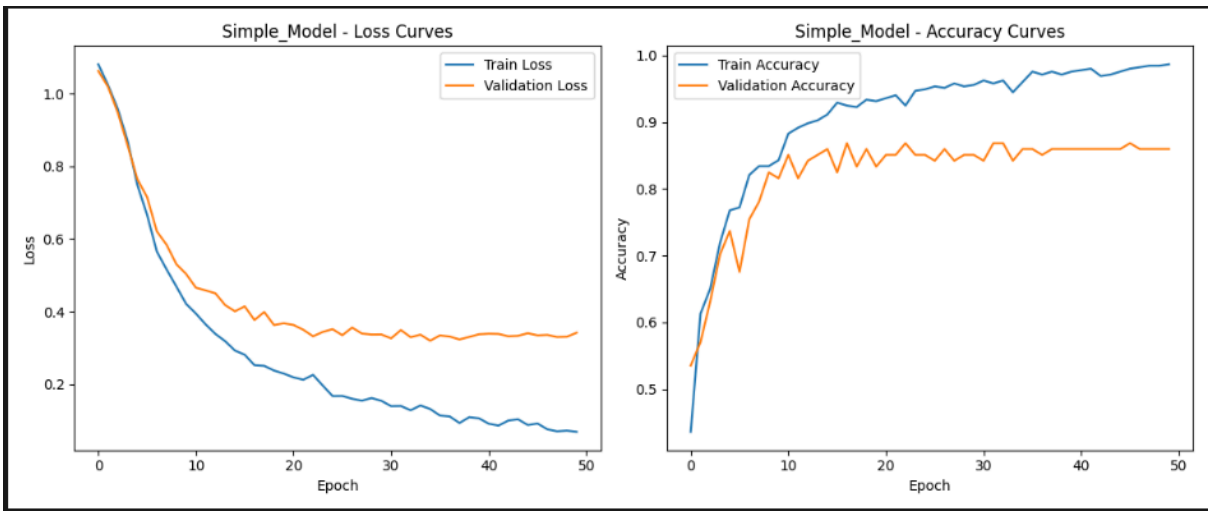
- **Prothom Alo**:
    - Lowest precision (0.77), recall (0.79), and F1-score (0.78).
    - Possible reasons: More dynamic content or visual similarities with others.

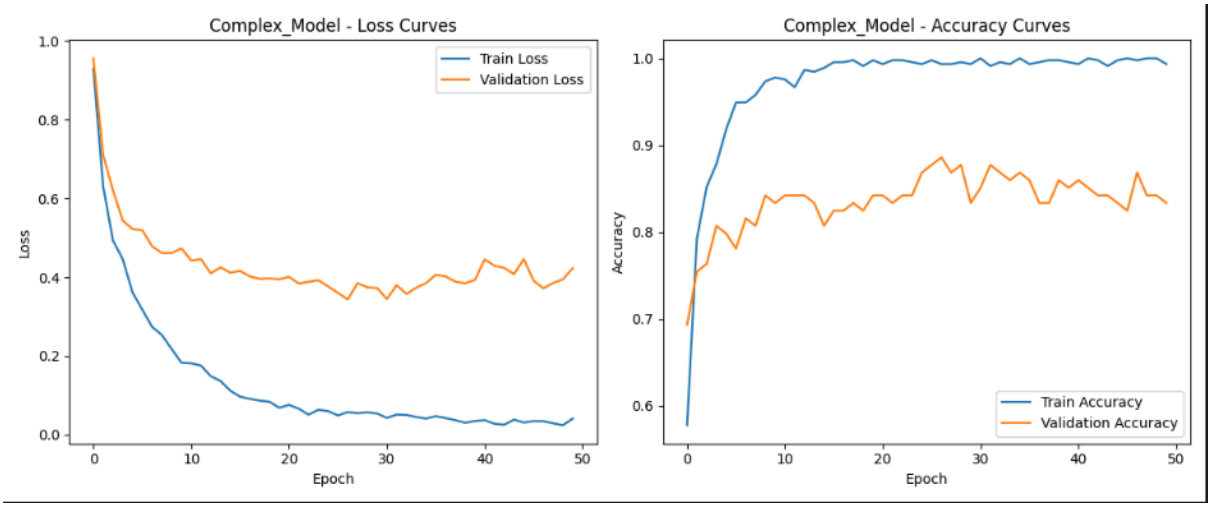

Simple_Model - Loss Curves
Train Loss
Validation Loss
Loss
Epoch
Simple_Model - Accuracy Curves
Train Accuracy
Validation Accuracy
Accuracy
Epoch


Complex_Model - Loss Curves
Train Loss
Validation Loss
Loss
Epoch
Complex_Model - Accuracy Curves
Train Accuracy
Validation Accuracy
Accuracy
Epoch

```
BATCH_SIZE = 32
EPOCHS = 50
LEARNING_RATE = 2e-4
TRAIN_SPLIT = 0.8
INPUT_SIZE = 188
HIDDEN_SIZE = 128
```

```
Final Evaluation of Complex Model:

Classification Report:
                              precision    recall  f1-score   support

https://cse.buet.ac.bd/moodle/      0.92      0.89      0.91        38
          https://google.com        0.92      0.92      0.92        38
       https://prothomalo.com       0.82      0.84      0.83        38

                    accuracy                           0.89       114
                   macro avg       0.89      0.89      0.89       114
                weighted avg       0.89      0.89      0.89       114


Model Comparison:
Simple Model Best Accuracy: 0.8684
Complex Model Best Accuracy: 0.8860
```




Complex_Model - Loss Curves
Train Loss
Validation Loss
Loss
Epoch
Complex_Model - Accuracy Curves
Train Accuracy
Validation Accuracy
Accuracy
Epoch