Cross-Manipulation Deepfake Detection with Vision-Language Foundation Models Report

Student Name: 陳祖佑

Student ID: 313832010

I. Methodology

Brief Introduction to CLIP: CLIP 為一種 VLM,由 OpenAI 開發,主要用於圖像與文字的聯合理解,核心理念為將圖像和文字嵌入到相同的向量空間,不僅僅是單純的分類。

將 Train_Real (Real_youtube, 10100 張圖)和 Train_Fake (FaceSwap, 10100 張圖)的部分當成訓練資料集,Real 部分設為 0;Fake 部分設為 1,因為是最後才做測試所以我先設定最終 ground truth 為 -1,事後才用計數方式去算準確度。

訓練使用的模型來自 transformers 的 CLIP (來自 openai),並使用 cuda=12.8 的 2.7.1 版 pytorch (torch)。 先載入 CLIPProcessor, CLIPModel 並設 定為推論(eval)模式。 模型功能為將圖片轉 RGB 格式後放入 CLIP 模型中並回 傳特徵。

正式訓練時,將總計 20200 張圖片依序放入模型中,並將特徵和 label 儲存。 隨後使用 scikit-learn 的 LogisticRegression 建立並訓練邏輯回歸模型(通常用於二元分類問題)。

訓練結束後進行測試,將 Test_Fake (NeuralTextures, 10100 張圖)當測試資料集,利用前面的模型進行 inference,得到機率,為了知道選擇是 0 還是 1,後面進行四捨五入。 測試的 f1-score 為 0.87421。

因設定預測 1 為 Fake, 而測試資料集全部都是 Fake, 當 Test_Pred 為 1 時代表預測正確。 經過判斷,有 77.653%的測資預測正確。

隨後將模型權重存入 pth 檔中 (clip_finetuned.pth),並將測資的預測結果 寫入 csv 中 (Results.csv)。

II. Experiments

II-A 基礎設定

由於訓練資料集較大 (超過 15GB,無法直接上傳或使用 Google drive 掛載雲端空間),所以後來移到本地端進行訓練(第一段第 16 行註解未刪用作如果用 Google colab 訓練)。 本地端使用的規格如下: (使用 GPU training, CPU 僅作為傳輸和部分 tensor 轉換)

CPU: R7 7840hs, GPU: RTX 4050 (6GB), RAM: 24GB, Python 3.13.2, torch = 2.7.1 + cu128.

Extra packages: transformers, numpy, os, PIL, tqdm, scikit-learn, pandas, csv.

訓練資料集: Real_youtube (Real, 10100 張圖), FaceSwap (Fake, 10100 張圖)

測試資料集: NeuralTextures (Fake, 10100 張圖)

Ⅱ-B 訓練/測試過程資訊

訓練執行時間(如下圖),耗費 12 分 54 秒。

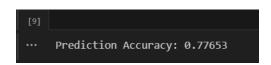


測試執行時間(如下圖),耗費6分26秒。

```
... 100%| 10100/10100 [06:26<00:00, 26.14it/s]
f1: 0.8742127849300563
```

II-C 最終結果

使用的方法如 Methodology 中所述,最終得到 77.653%準確度(如下圖)。



III. Discussion

1. 為何程式碼中沒有 EER 或 ROC curve 等?

A: ROC curve 和 ROC AUC 需要測試資料集中有 positive (real, 0)和 negative (fake, 1)同時存在,然而測試資料集全由 fake (1)組成,所以算不出來。 而 EER (Equal Error Rate 也依賴 TPR/FPR),故也算不出來。

2. 為何沒使用 LoRA 等技術進行 fine-tuned?

A: LoRA 為透過風格融入來進行微調,但資料集中圖片雖都是人在講話的畫面,但未必可透過特定風格去微調。 所以最後沒採用。

3. 一開始遇到的難題:

A: Google Colab 無法上傳如此大的資料集(上傳失敗),也沒辦法透過雲端硬碟掛接來解決(超過 15GB, google drive 每人只有 15GB 免費限度),於是改用本地端的 VScode 執行。

本地端因版本原因有可能 Clip 不支援 2.7.1 的 torch,需使用 use_safetensors = True,雖然可能效能不再是最佳,但能正常運行程式。

4. 最後遇到的難題:

A: 在首次執行後隔天開機發現整個環境都被毀掉,環境需要進行重建,因為耗費很多時間,環境重建後只針對特定小區域的問題進行了修改。

5. 結果討論

A: 以結果而言 77.653%準確度在沒有任何 fine-tuning 的情況下算普通,測資中可能存在偏差問題,因為所有測資答案皆為 1 (Fake)。

V. References

1. https://arxiv.org/abs/2403.16442

Topic: If CLIP Could Talk: Understanding Vision-Language Model Representations
Through Their Preferred Concept Descriptions.

2. https://github.com/openai/CLIP

Topic: openAI / CLIP

VI. Related Links (HW's repository)

Github repository: https://github.com/Pystarter0/Module_Course_AI-_HW

VII. Extra: Screenshot during Execution

