Qui connait CLIP ?

Ryan WEBSTER Teddy FURON

Centre Inria de l'Université de Rennes, Campus de Beaulieu, 35150 Rennes, France

Résumé – Les modèles multimodaux sont entrainés sur des milliards d'images provenant d'Internet qui contiennent des visages humains et des descriptions d'individus. Ces modèles posent donc des problèmes de protection de la vie privée. Ce travail formalise le problème de l'extraction d'identité, dans lequel un attaquant peut extraire de manière fiable les noms des individus connu d'un modèle multi-modal. Nous montrons comment l'extraction d'identité peut être utilisée pour vérifier la confidentialité des modèles.

Abstract – Multi-modal foundational models are trained on billions-scale training Internet images which contain human faces and descriptions of individuals. Thus, these models pose potentially widespread privacy issues. This work formalizes the problem of identity extraction, wherein an attacker can reliably extract the names of individuals known by a multi-modal model. We show how identity extraction can be used to audit model privacy.

1 Introduction

Levons d'abord l'ambiguïté du titre accrocheur. La question n'est pas de savoir si le lecteur connaît CLIP, un modèle multi-modal image / texte, mais bel et bien quelles sont les personnes connues de CLIP. Un modèle CLIP étant une composante essentielle des IA génératives d'images, on répond en fait à la question "Quelles personnes sont 'générables' par une IA ?".

Un modèle CLIP est appris à partir de milliards de paires (image / description) scrappées sur Internet, ce qui pose des problèmes de vie privée. Les fournisseurs d'IA générative en sont conscients [15] puisqu'ils interdisent les requêtes demandant les noms des personnes sur une photo ou la génération d'images représentant des personnes. Le nouveau jeu d'entraînement `DataComp` floute les visages détectés sur les images.

Il y a peu de travaux sur l'audit a posteriori d'un modèle multi-modal. D. Hintersdorf *et al.*ont écrit le papier "*Does CLIP know my face ?*" [8] présentant une attaque d'appartenance (*membership inference attack*). Il ne s'agit pas de savoir si une donnée précise a servi à l'entraînement du modèle CLIP, mais si des données (potentiellement inconnues) relatives à une personne précise ont servi. Cette attaque prend en entrées des images d'une personne ainsi que son véritable nom.

Notre travail étend l'article pionner [8] en proposant non pas une attaque d'appartenance mais une **attaque d'extraction d'identité** : on souhaite utiliser un modèle CLIP pour extraire les noms de personnes sur des photos. On suppose que le modèle est capable d'identifier une personne s'il l'a "vue" pendant son d'entrainement. Le cas échéant montre que ce modèle CLIP ne connait pas cette personne car le jeu d'entrainement ne contient aucune donnée relative à cette personne.

Ce problème est difficile car il est impossible d'établir une vérité terrain. Les modèles sont appris sur des milliards de données. Les noms des personnes apparaissent souvent avec des orthographes approximatives ou par des paraphrases comme "Le président de la République Française" (qui, suivant la date de l'image associée, n'est pas forcément E. Macron).

2 Bibliographie

Modèles multi-modaux Les modèles CLIP [16, 11, 19] évaluent un score de vraisemblance entre une image et un texte : est-ce que ce texte est une description vraisemblable de cette image ? Ces modèles sont appris sur des jeux contenant des milliards de données comme `LAION-2B` ou `DataComp` [17, 6].

Attaque d'appartenance ou d'extraction Une attaque d'appartenance MIA[1] révèle si une donnée précise a été utilisée pour entraîner un modèle [18, 2, 7, 5, 10]. Cette attaque a été menée sur des modèles CLIP [13]. Une attaque IMIA[2] révèle si des données (images, noms, pseudos, ...) concernant une personne ont servi à l'entraînement. Cette attaque est menée contre des IA générative directement [21] ou indirectement contre un modèle CLIP composante d'une IA générative [8]. Une attaque d'extraction recouvre partiellement des données d'entraînement à partir d'un modèle, qu'il soit un LLM [3, 14] ou un générateur d'images [4, 20].

Cet article combine les deux dernières attaques en une nouvelle IEA[3] : il s'agit d'une extraction d'identité (et non pas de données d'entraînement précises).

3 Énoncé du problème

Soit z une identité, élément de l'ensemble $\mathcal{Z}$. Cette identité a pour nom $n_z \in \mathcal{N}$. On note une image par X et un texte (description) C. L'ensemble $\mathcal{X}_z$ contient des images représentant l'identité z et $\mathcal{C}_z$ des textes incluant son nom n_z.

Soit X_z une image prise au hasard dans $\mathcal{X}_z$ et x_z une image particulière de cet ensemble. Les algorithmes présentés ici sont appelés *attaques* et les personnes les exécutant des *attaquants* par tradition. L'indice A note les ensembles disponibles à l'attaquant (e.g. $\mathcal{Z}^A$) et l'indice T les ensembles utilisés à l'entrainement du modèle M (e.g. $\mathcal{Z}^T$).

[1] *Membership Inference Attack*
[2] *Identity Membership Inference Attack*
[3] *Identity Extraction Attack*

Hypothèses On suppose un scénario 'boite noire' où l'attaquant ne peut que requêter le modèle M avec une paire (image X, texte C) et reçoit le scalaire $M(X, C)$ appelé 'score CLIP'.

Attaque d'appartenance d'identité - IMIA Une IMIA révèle si les données d'une personne ont servi à l'entrainement du modèle M. C'est un test d'hypothèse à propos de l'identité z. L'hypothèse nulle $\mathcal{H}_0$ est qu'aucune donnée relative à z n'a servi pour entraîner M. L'hypothèse alternative $\mathcal{H}_1$ est que l'entrainement a utilisé des données de z, *i.e.* des paires (X_z, C_z). Ici, l'emploi de lettres capitales montre que l'attaquant ne connait pas ces données exactement. Une IMIA est donc une généralisation d'une MIA qui est un test sur l'appartenance de la donnée précise (x_z, c_z) au jeu d'entraînement. L'attaquant a son propre ensemble d'identité à tester $\mathcal{Z}^A$ qui peut être en intersection non nulle avec l'ensemble $\mathcal{Z}^T$ des identités dont des données ont servi à l'entraînement de M.

Attaque d'extraction d'identité - IEA Dans notre nouveau défi, l'attaquant n'a qu'un ensemble d'images $\mathcal{X}^A$, union de sous-ensembles $\mathcal{X}_z^A$. Chaque image $X \in \mathcal{X}_z^A$ représente une unique entité z inconnue de l'attaquant. Une solution est de mener une IMIA sur chaque sous-ensemble $\mathcal{X}_z^A$ avec un grand jeu de noms $\mathcal{N}^A$ avec l'espoir que $n_z \in \mathcal{N}^A$. Une autre solution, que l'on appellera extraction par génération, est de faire en sorte qu'un générateur de nom trouve n_z. Une difficulté majeure de l'IEA est que l'attaque doit identifier z par son nom jusqu'alors inconnu, contrairement à une IMIA.

4 Attaques

Cette section décrit d'abord l'attaque IMIA de l'état de l'art avant de présenter notre attaque IEA.

4.1 IMIA contre CLIP

Partant d'une identité z, l'attaquant réunit d'abord dans l'ensemble $\mathcal{X}_z^A$ des images de cette personne. Il possède aussi K prototypes $\{\mathsf{C}^k\}_{k=1}^K$ pour écrire des descriptions d'image. Par exemple, appliqué au nom n_z ='John Doe', cela donne des descriptions $C_z = \mathsf{C}^k(n_z)$ comme 'A photo of John Doe', 'John doe in a suit', ...L'attaquant soumet au modèle CLIP des paires correctes $(X_z, \mathsf{C}^k(n_z))$ et des paires fausses $(X_z, \mathsf{C}^k(n_{z'}))$ avec $n_{z'} \in \mathcal{N}^A$ comme expériences témoins. L'intuition est que la paire correcte reçoit un score plus élevé si $z \in \mathcal{Z}^T$. L'attaque identifie alors la personne sur l'image X_z en prenant l'argument maximum des scores CLIP (1). Cette prédiction est faite par image et par prototype. Elle est agrégée en une identification $\hat{z}\,(k)$ par prototype au moyen d'un vote majoritaire (2) sur l'ensemble des images $\mathcal{X}_z^A$. La probabilité de succès est estimée par la fréquence du vrai nom de la personne sur l'ensemble des prototype (3).
$\forall X_z \in \mathcal{X}_z^A, \forall k, 1 \leq k \leq K$:

$$\hat{n}_z(X_z, k) = \arg \max_{n \in \mathcal{N}^A \cup \{n_z\}} M\left(X_z, \mathsf{C}^k(n)\right), \quad (1)$$

$$\hat{n}_z(k) = \mathsf{MajorityVote}_{X \in \mathcal{X}_z^A}\left(\hat{n}_z(X, k)\right), \quad (2)$$

$$f_z = \mathsf{Average}_{1 \leq k \leq K}\left(\mathbb{1}[\hat{n}_z(k) = n_z]\right). \quad (3)$$

L'identité z est finalement considérée comme faisant partie de l'entrainement si $f_z > \tau$, où τ est un seuil à fixer. Sous l'hypothèse $\mathcal{H}_0$, l'identité z ne fait pas partie de l'entrainement et le nom estimé $\hat{n}_z(k)$ (2) est aléatoire, en espérance, $f_z = 1/|\mathcal{N}^A|$. Cet modèle est valide si $\mathcal{N}^A$ contient des identités "*that are culturally similar to*" z [8] pour éviter des biais dans la distribution de $\hat{n}_z(k)$ sous $\mathcal{H}_0$.

4.2 IEA contre CLIP

Dans une extraction, l'attaquant a un ensemble d'images et un ensemble de noms, mais il ignore les associations. On suppose ici que son ensemble de noms $\mathcal{N}^A$ a une intersection non vide avec la vérité terrain.

La difficulté principale est la construction de l'ensemble des noms 'suspects' $\mathcal{N}^A$. L'idée est de générer une partie de cet ensemble à partir des images $\mathcal{X}_z^A$ guidé par le score CLIP. Nous considérons deux méthodes décrites ci-dessous, l'une requête un modèle VLM (Vision Language Model) de description d'image (captioning), l'autre affine un LLM pour qu'il devine le vrai nom de la personne.

Un grand nombre de noms distracteurs (10 000) sont ensuite ajoutés à cette liste pour donner l'ensemble $\mathcal{N}^A$ final. L'attaquant calcule ensuite les fréquences (3) pour chacun de ces noms 'suspects' à partir des scores CLIP sur les images du sous-ensemble $\mathcal{X}_z^A$. Il identifie $\hat{z}$ comme étant l'identité ayant la plus grande fréquence : $\hat{z}^\star = \arg\max_{z':n_{z'} \in \mathcal{N}^A} f_{z'}$. Un garde-fou est de refuser d'identifier, $\hat{z} = \emptyset$, si cette fréquence maximale est trop petite ou si l'écart avec le second maximum est trop faible. Cette dernière étape est importante pour filtrer les hallucinations et améliorer la précision de l'attaque.

VLM L'attaquant requête le VLM avec une ou des images $\mathcal{X}_z^A$ et extrait des noms dans les descriptions retournées. Ceci est fait par sous-ensemble d'images $\mathcal{X}_z^A$. Si le VLM ne connait pas la personne, il hallucine. Les fréquences estimées sont toutes faibles et de même niveau. Une identification infructueuse est due au fait que cette identité est inconnue du VLM ou du modèle CLIP.

Les VLMs utilisés sont GPT-4o, Qwen2-72B, Gemini-1.5-Pro et Pixtral-12B. Certains refusent de nommer des individus sur des images. Nous avons donc dû contourner cette mesure de sécurité. Le coeur des prompts reste générique et neutre comme "*Describe the subject of this artwork*". Nous ne soumettons au VLM qu'une seule image par identité.

LLM Cette deuxième approche affine un modèle de langue $P_{\theta^\star(z)}$ pour qu'il produise des descriptions vraisemblables pour toutes les images représentant l'identité z :

$$\theta^*(z) = \arg\max_{\theta \in \Theta} \mathbb{E}_{C \sim P_\theta}\left[\sum_{X \in \mathcal{X}_z^A} M(X, C)\right] \quad (4)$$

Résoudre (4) n'est pas trivial. Le modèle CLIP étant en boite noire, ni les descriptions CLIP internes ni le gradient du CLIP score sont accessibles. Nous utilisons la méthode REINFORCE [22] récemment revisitée pour les LLMs [1]. Le LLM est initialisé par un modèle open-source comme Mistral-7B [12]. A chaque itération, le LLM génère un lot de descriptions dont les score CLIP sont calculés en rapport avec les images de $\mathcal{X}_z^A$ puis sommés (4). La méthode REINFORCE estime le gradient de manière empirique et met à jour les paramètres du modèle θ.

L'affinage du LLM se fait en LoRa [9] avec la méthode REINFORCE pour chaque identité testée. Nous utilisons 50 images par identité. Nous ajoutons aussi des images distracteurs : 1 000 images d'autres identités choisies au hasard. Leurs CLIP scores reçoivent une pondération négative avant d'être ajoutés à (4). La température du LLM est relativement haute, $T = 1.25$, pour promouvoir la diversité dans la génération. Nous demandons au moins 24 tokens par description et l'algorithme fait 50 itérations.

5 Partie expérimentale

5.1 Protocole

Cette section détaille les jeux de données et les modèles.

Images et noms Pour l'attaquant, nous avons choisi VGGFace2 pour sa taille (8 000 identités), sa diversité (égalité homme/femme, plus d'un quart de personnes non Caucasiennes, poses différentes). L'attaquant a aussi besoin de noms 'distracteurs'. Nous avons rassemblé tous les noms de Wikipedia (version anglaise) répertoriés sous la caétgorie '*Living people*'. Les noms communs avec VGGFace2 sont retirés, pour faire un ensemble d'un millions de distracteurs. Pour chaque attaque, nous tirons au hasard 10 000 noms dans cet ensemble.

Modèles Les modèles CLIP sont ceux de OpenCLIP [11] qui sont appris sur `LAION-2B` (L2B) ou `DataComp` (DC). Leur nomenclature indique leur architecture et leur jeu d'entrainement : ViT-H-14_L2B, ViT-H-14_DC ...

Reconstitution d'une vérité terrain Nous calculons l'intersection des identités de VGGFace2 et des noms apparaissant dans les descriptions de `LAION-2B` ou `DataComp`. Cela demande un certain travail non décrit ici pour normaliser l'orthographe des noms et pour repérer les identités nommées dans `LAION-2B` ou `DataComp`. Finalement, on ne retient que les noms qui apparaissent au moins dix fois dans le jeu d'entrainement. La figure 1 montre l'histogramme du nombre d'occurences dans `LAION-2B`.

Métriques Autant les métriques pour une IMIA [8] sont évidentes (taux de faux positif, taux de faux négatif), autant les métriques d'une IEA sont à inventer.

Nous proposons le critère suivant : Sous $\mathcal{H}_1$, l'attaque est réussie si i) elle donne un nom, ii) si ce nom a une distance de Levenshtein de 1 avec le nom de la vérité terrain. Par exemple, *Odonnell* pour *O'Donnell* est une extraction réussie. Nous mesurons combien de noms sont extraits et calculons le taux d'extraction (pourcentage) par rapport au nombre d'identités à trouver dans les images $\mathcal{X}^A$. La deuxième métrique notée E@95 est ce même pourcentage quand 95% des noms extraits sont effectivement corrects. On se concentre ici sur des taux d'erreurs faibles. Il fait sens d'être certain d'avoir extrait les bons noms plutôt que de chercher à en extraire un maximum.

Resultats quantitatifs Le Tableau 1 montre les résultats de l'extraction sur 4 000 identités de VGGFace2. L'attaque VLM retourne une dizaines de noms tandis que l'attaque LLM en produit 50. Un grand nombre de noms distracteurs (10 000) sont ensuite ajoutés à cette liste et leurs fréquences sont calculées. Cette dernière étape est importante pour filtrer les hallucinations et améliorer la précision de l'attaque. Sans cette dernière étape, la précision de l'attaque est bien moindre dû aux hallucinations (cf. Fig. 2).

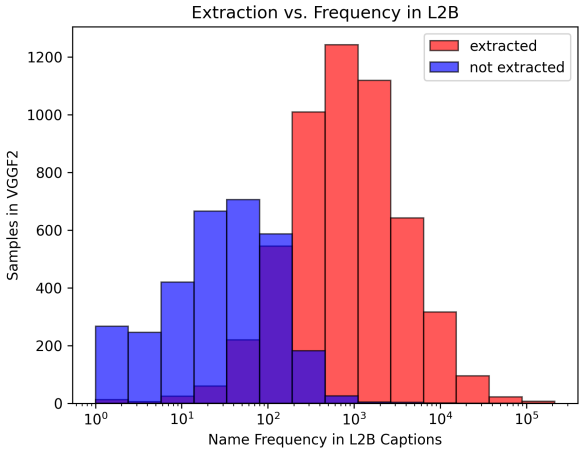

FIGURE 1 : Nombre d'identités extraites du modèle CLIP ViT-B-32_L2B en fonction de la prévalence de leurs noms dans le jeu d'entrainement `LAION-2B`.

TABLE 1 : Performance des attaques d'extraction.

Modèle	Taux d'extraction	E@95
LLM+REINFORCE		
ViT-L-14_L2B	12.68	7.53
ViT-H-14_L2B	11.48	7.45
Convnext-xxlarge_L2B	11.8	7.15
ViT-H-14_DC	9.17	5.24
ViT-B-32_L2B	11.48	7.50
VLM (Vision-Language Models)		
GPT-4o	52.51	22.25
Qwen2-72B	22.83	13.06
Pixtral-12B	8.40	4.93
Gemini-1.5-Pro	33.48	19.34

Analyse qualitative Grâce à la méthode REINFORCE, le LLM génère des descriptions de plus en plus pertinentes. Par exemple, la profession de l'entité est rapidement trouvée même si son nom n'est toujours pas incorrect.

L'extraction à base de VLM est parfois très efficace. Par exemple, GPT-4o trouve un nom dans $\approx 52\%$ des cas, même avec une seule image. Ceci dit, ce VLM est bien plus gourmand en temps de calcul que la méthode REINFORCE. Toutefois, un 'petit' VLM comme Pixtral-12B est moins performant que l'attaque LLM avec REINFORCE.

Audit de vie privée L'extraction de l'identité est plus hasardeuse sur les modèles CLIP entrainés sur `DataComp` où les visages ont été floutés pour protéger la vie privée [6]. La métrique E@95 va de 36.3% pour ViT-L-14_DC à 6.1% pour ViT-B-16-quickgelu_DC. Il est donc toujours possible d'extraire des noms avec précision ce qui montre que le floutage opéré sur `DataComp` n'est pas suffisant.

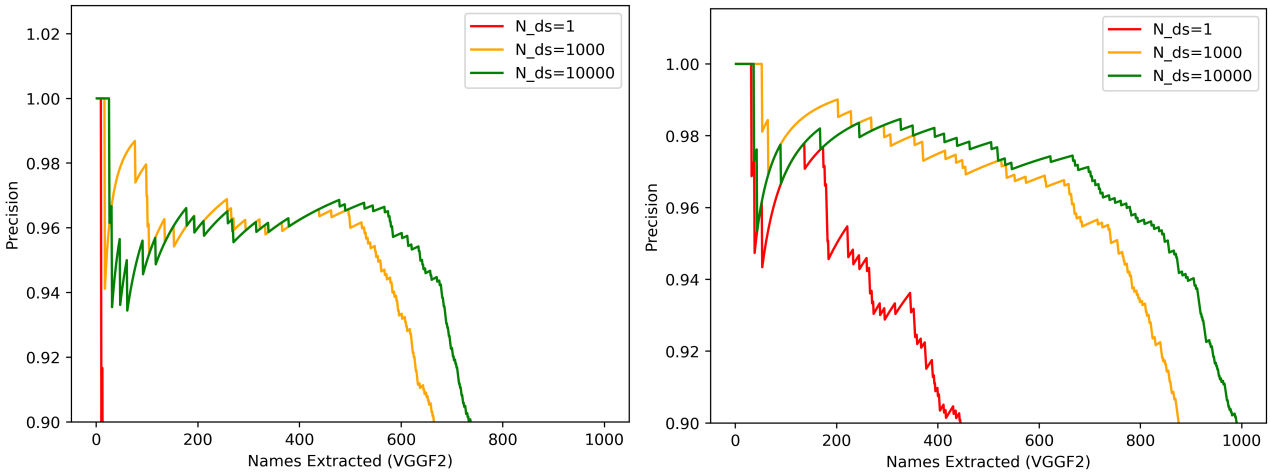

FIGURE 2 : De l'importance de filtrer les résultats des VLMs Qwen2-72B (gauche) et Gemini-1.5-Pro (droite) en rajoutant des noms distracteurs (1, 1 000 ou 10 000). Modèle CLIP ViT-B-32_L2B.

6 Conclusion

Ce papier étend les attaques d'appartenance d'identité au concept d'extraction d'identité à partir d'un modèle CLIP. Le succès de ces attaques d'extraction montre que CLIP peut jouer le rôle d'un système de reconnaissance faciale, non seulement pour la vérification mais aussi pour l'identification, et ce, même si les noms ne sont pas connus. Les attaques proposées sont utiles pour faire un audit du respect de la vie privée.

Références

[1] Arash AHMADIAN *et al.* : Back to basics : Revisiting reinforce-style optimization for learning from human feedback in llms. *In ACL*, 2024.

[2] N. CARLINI, S. CHIEN, M. NASR, S. SONG, A. TERZIS et F. TRAMER : Membership inference attacks from first principles. *In IEEE S&P*, 2022.

[3] N. CARLINI *et al.* : Extracting training data from large language models. *In USENIX Security 21*, 2021.

[4] N. CARLINI, J. HAYES, M. NASR, M. JAGIELSKI, V. SEHWAG, F. TRAMER, B. BALLE, D. IPPOLITO et E. WALLACE : Extracting training data from diffusion models. *In USENIX Security*, 2023.

[5] Dingfan CHEN, Ning YU, Yang ZHANG et Mario FRITZ : GAN-leaks : A taxonomy of membership inference attacks against generative models. *In ACM CCS*, 2020.

[6] Samir Yitzhak GADRE *et al.* : Datacomp : In search of the next generation of multimodal datasets, 2023. arXiv 2304.14108.

[7] J. HAYES, L. MELIS, G. DANEZIS et E. DE CRISTOFARO : LOGAN : Membership inference attacks against generative models. *PET Symposium*, 2019.

[8] D. HINTERSDORF, L. STRUPPEK, M. BRACK, F. FRIEDRICH, P. SCHRAMOWSKI et K. KERSTING : Does CLIP know my face ? *J. Artif. Int. Res.*, 2024.

[9] E. J HU, Y. SHEN, P. WALLIS, Z. ALLEN-ZHU, Y. LI, S. WANG, L. WANG, W. CHEN *et al.* : LoRa : Low-rank adaptation of large language models. *ICLR*, 2022.

[10] Hailong HU et Jun PANG : Membership inference of diffusion models. 2023. arXiv 2301.09956.

[11] G. ILHARCO *et al.* : OpenCLIP. *Zenodo*, 2021.

[12] Albert Q. JIANG *et al.* : Mistral 7B, 2023. arXiv 2310.06825.

[13] M. KO, M. JIN, C. WANG et R. JIA : Practical membership inference attacks against large-scale multi-modal models : A pilot study. *In ICCV*, 2023.

[14] M. NASR *et al.* : Scalable extraction of training data from (production) language models, 2023. arXiv 2311.17035.

[15] OPENAI : CLIP : connecting text and images, 2021. `https://openai.com/index/clip/`.

[16] A. RADFORD *et al.* : Learning transferable visual models from natural language supervision. *In ICML*, 2021.

[17] C. SCHUHMANN *et al.* : LAION-5B : An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022.

[18] Reza SHOKRI, Marco STRONATI, Congzheng SONG et Vitaly SHMATIKOV : Membership inference attacks against machine learning models. *In IEEE S&P*, 2017.

[19] Quan SUN, Yuxin FANG, Ledell WU, Xinlong WANG et Yue CAO : EVA-CLIP : Improved training techniques for CLIP at scale, 2023. arXiv 2303.15389.

[20] Ryan WEBSTER : A reproducible extraction of training images from diffusion models. 2023. arXiv 2305.08694.

[21] Ryan WEBSTER, Julien RABIN, Loic SIMON et Frederic JURIE : This person (probably) exists. Identity Membership Attacks against GAN generated faces, 2021. arXiv 2107.06018.

[22] Ronald J. WILLIAMS : Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256, 1992.