

Imperial College London
Department of Computing

Online Identity Inference, Impersonation and Mitigation Strategies

Anastasios Lepipas

July 2025

Supervised by: Dr. Soteris Demetriou

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing of Imperial College London and
the Diploma of Imperial College London

Statement of Originality

I, Anastasios Lepipas, declare that the thesis entitled ‘Online Identity Inference, Impersonation and Mitigation Strategies’ and the work presented in the thesis are both my own, and have been generated by me as the results of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at Imperial College London,
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;

Copyright Declaration

I herewith certify that the material in this thesis that is not my own work has been properly acknowledged and referenced.

Anastasios Lepipas.

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose. This on the condition that; you credit the author and do not distribute modified versions of the work.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Abstract

Online identities in social platforms and generative models are increasingly at risk as adversaries exploit unique identifiers for malicious ends. This thesis examines two complementary perspectives on such vulnerabilities: squatted usernames on social networks and attribute leakage in text-guided diffusion (T2I) models.

First, we systematically investigate how attackers capitalize on near-duplicate username registrations on X, confusing users and enabling large-scale impersonation. To illuminate this phenomenon, we develop the username-generation tool UsernameCrazy, inspecting hundreds of thousands of variants derived from celebrity accounts. Our measurements reveal a concerning ecosystem of suspended and active squatted usernames, many featuring profile pictures and names closely matching the originals to bolster deception. We further demonstrate how user mismentions and X’s search algorithm magnify these threats. To mitigate them, we propose SQUAD, a framework integrating UsernameCrazy, achieving a 94% F_1 -score in detecting suspicious squatted accounts.

While username squatting targets explicit digital identities, online personas are increasingly shaped by more subtle signals—particularly in the era of generative AI. We extend our analysis to emerging T2I models and their potential to leak sensitive traits—such as authorship or dementia status—through seemingly benign image outputs. By constructing adversarial pipelines that leverage image augmentation and text-image embedding models, we achieve up to 0.877% Top-5 accuracy in attributing images across 100 authors, and 0.75% accuracy in inferring dementia status (using the ADReSS dataset). These inferences are robust against diverse training sets, independent of classifier choice, and remain resilient to standard mitigation strategies.

Together, these findings reveal how adversaries can hijack identifiers—whether user-chosen names on a social platform or latent attributes embedded in generative images—to threaten security and privacy. By characterizing these overlooked risks and introducing detection and protection mechanisms, this thesis advances our collective understanding of modern identity vulnerabilities and highlights avenues to safeguard online communities.

Dedication

This PhD thesis is dedicated to my beloved mother and father who have dedicated their life to make me the person I am and have always supported me through bad times. It's their support, love and encouragement that made it possible for me to conduct research at the highest level possible. My PhD is also dedicated to my one and only brother who is simply the most important thing I have in this world. Mom, Dad and Brother I love you!

Acknowledgements

A great deal of support has been poured into the culmination of this thesis. At this point, I would like to thank all the people that supported me throughout my challenging and exciting PhD journey.

My deepest gratitude goes to my supervisor, mentor, and friend, Dr. Soteris Demetriou, whose unwavering support has been priceless to me both academically and personally throughout this challenging journey. Working with Soteris has been an immense honor. This thesis is a testament to his tireless assistance and guidance, and the countless hours he dedicated to helping me succeed. Without his encouragement and aid, this accomplishment would not have been possible. Soteris, I enjoyed the conversations we had about Greek/Cypriot culture and of course science!

Most importantly, I would like to thank my family for their unwavering support. In particular my parents, I will never be able to thank them enough for their guidance and unconditional love. And to my awesome sibling, who has always been there for me.

I would also like to express my gratitude to all my friends in the APSS lab for the research discussions and helpful advice. Being part of the APSS lab has been a wonderful experience, thanks to the exceptional people who make it so.

Nomenclature

Acronyms

AA Authorship Attribution

BERT Bidirectional Encoder Representations from Transformers

BLIP Bootstrapping Language-Image Pre-training

CLIP Contrastive Language-Image Pretraining

CNN Convolutional Neural Network

DP Differential Privacy

GloVe Global Vectors for Word Representation

GPT Generative Pre-trained Transformer

LLM Large Language Model

NLI Natural Language Instruction

OSN Online Social Networks

T2I Text-to-Image

ViT Vision Transformer

Symbols

$\mathbb{G}$	Embedding Function
$\mathbb{P}$	Input Prompts
$\mathbb{S}$	Attribute Classes
$\mathcal{A}$	Adversary
$\mathcal{A}_{io}^d$	Dementia Multi-Modal Input-Output Inference
$\mathcal{A}_i^d$	Dementia Input Inference
$\mathcal{A}_o^d$	Dementia Output Inference
$\mathcal{A}_{io}$	Multi-Modal Input-Output Inference
$\mathcal{A}_i$	Input Inference
$\mathcal{A}_o$	Output Inference

Contents

Abstract	v
List of Figures	xix
List of Tables	xxiii
1 Introduction	1
1.1 Overview	1
1.2 Thesis outline, state-of-the-art and contributions	3
1.2.1 Thesis Outline	3
1.2.2 Related Work	4
1.2.3 Thesis Contribution	5
1.3 Dissemination	8
2 Background Work	11
2.1 Online Identity	11
2.2 Fake Accounts	13

2.3	Harmful Content	15
2.4	Property Inference Attacks	18
2.5	Embedding Models	21
2.6	Summary	23
3	Related Work	25
3.1	Account Threats on Social Networks	25
3.2	Fake Content on Social Networks	28
3.3	Squatting Techniques and Tools	29
3.4	Authorship Inference & Obfuscation	30
3.5	Text-to-Image Diffusion Models	33
3.6	Image Captioning	37
3.7	Dementia Classification	38
3.8	Summary	39
4	Measuring Username Squatting on X	41
4.1	Problem Statement	42
4.2	Measurement Study Design	45
4.2.1	Ethics	45
4.2.2	Data Collection.	46
4.2.3	Username Generation (UsernameCrazy).	46
4.2.4	Code Availability	50

4.3	Measurement Methods	50
4.3.1	MQ1. Measuring Prevalence of Username Squatting	51
4.3.2	MQ2. Measuring Squatted Account Effects:	52
4.3.3	MQ3. Measuring Variant Characteristics	53
4.3.4	MQ4. Measuring Account Behaviour	53
4.4	Characterization	56
4.4.1	Prevalence of Username Squatting.	57
4.4.2	Online Confusion.	59
4.4.3	Squatted Username Characteristics.	61
4.4.4	Squatted Account Behaviour.	63
4.5	Summary	66
5	An Impersonation Detection Framework for X: Design and Evaluation	69
5.1	SQUAD Design Overview	69
5.2	Feature Extraction and Selection	71
5.3	Dataset Generation	74
5.4	Evaluation of SQUAD	75
5.4.1	UsernameCrazy Performance	76
5.4.2	Classification Performance of SQUAD.	80
5.5	Applying SQUAD	84
5.5.1	Discussion	85

5.5.2	Responsible Disclosure.	87
5.5.3	Code Availability	88
5.6	Conclusion	88
6	Attribute Leakage in Text-To-Image Models	91
6.1	Problem Statement	92
6.2	Black-Box Attribute Inference Attacks	98
6.2.1	Threat Model and Framework Overview	98
6.2.2	Input Attack ($\mathcal{A}_i$)	103
6.2.3	Output Attack ($\mathcal{A}_o$)	104
6.2.4	Multi-Modal Input-Output Attack ($\mathcal{A}_{io}$)	106
6.3	Evaluation Setup	107
6.4	Evaluation	110
6.4.1	Attack Effectiveness	110
6.4.2	Leakage Analysis	119
6.4.3	Model Diversity	122
6.4.4	Leakage of other attributes	124
6.4.5	Code Availability	128
6.5	Robustness Against Obfuscation	129
6.5.1	Setup and Mitigation Approaches	129
6.5.2	Results	131

6.5.3	Directions for Improved Mitigation	135
6.5.4	Conclusion	140
7	Conclusion	143
7.1	Summary of Thesis Achievements	143
7.1.1	Future Research Directions	147
	Bibliography	151

List of Figures

3.1	Diffusion models perturb the data by progressively injecting noise, and then reconstruct fresh data by running the procedure in reverse. During this backward phase, each denoising step involves estimating the corresponding transition kernel. Figure taken from Jiang et al. [148].	35
4.1	Typo-mention: <i>CNN</i> mistakenly mentions <i>@cnnnbrk</i> , a squatted user of the official <i>@cnnbrk</i> profile.	44
4.2	Impersonation Attempts.	45
4.3	Taxonomy of UsernameCrazy’s generation models. The <i>blue-shaded</i> boxes indicate models which are not supported by existing tools. In <i>boldface</i> are username examples which cannot be generated by existing tools.	49
4.4	Squatted accounts in the top search recommendations grouped by edit distance.	62
4.5	The number of active and suspended accounts across different edit distance categories.	63
4.6	Number of actual tweets that mention at least one username variant grouped by the variants’ edit distance with their original accounts.	64

4.7	Classification of the confusing accounts.	67
5.1	High-level architecture of SQUAD.	70
5.2	The effectiveness of a) each generation model with <i>self-repetition</i> and b) <i>model stacking</i> . In <i>parenthesis</i> is the effectiveness of the primitive models—not shown on the bars.	79
5.3	ROC Curve of all the classification models.	83
6.1	Privacy leakage in Text-to-Image model usage. Users create natural language instructions which are used to guide the generation of images. Users share their instructions and/or the generated images with a false expectation of privacy online and by default with online model providers. Mallory who observes the shared data, aims to infer sensitive attributes about the data owner. This Figure depicts the scenario of authorship inference. Disclaimer: the images at Step 2 were generated solely for the purpose of illustration using <i>craiyon</i> [62] which at the time of writing was available for free online and anonymous usage.	93
6.2	High-level taxonomy of attribute (s^*) inference attacks against text-to-image models. We assume a hierarchical adversary with access to different information of the text-to-image model: (1) $\mathcal{A}_i$ infers the attribute from the input prompts P only, (2) $\mathcal{A}_o$ infers the attribute from the output images I and (3) $\mathcal{A}_{io}$ combines both the input and the output of the target model for inference.	99
6.3	Inference Framework Overview.	101
6.4	CDF of authors by minimum prompt count.	109
6.5	Comparison of all adversary models on DiffusionDB.	113

6.6	Analysis on 50 authors with 50 NLI's each (a,b) and 50 images generated from those NLI's using DiffusionDB dataset.	120
6.7	Author's Prompt: " <i>pig in a red sweater flying in a cyber-punk city</i> ". A_o : Top-1 = α_{42} (in Top-5).	121
6.8	CDF of CLIP scores for 10K images per model.	124
6.9	Two examples of transcriptions of <i>AD</i> patients describing the Cookie Theft picture shown.	127
6.10	The attack success rate of static and adaptive adversarial InferSent-based models against LLMs and PEGASUS. The defense method is shown in parentheses.	132
6.11	CDF of CLIP scores for 500 images generated from original NLI's (Stable Diffusion v2.1), and from paraphrased NLI's using different obfuscation mechanisms.	135
6.12	Qualitative comparison of end-to-end utility of mitigation strategies. T2I images, their NLI and text similarity score in parenthesis with the original NLI. (a) shows the original image included in DiffusionDB; (b) the original NLI is given as input to Stable Diffusion v2.1 [268]; in (c), (d), (e), (f), (g), (h) and (i) the input is the paraphrased NLI with different temperatures ($T = 1, 1.5, 2, 3, 5, 10$ and 15). In (j) and (k) we use the paraphrases from Llama3 and Mistral. The full text produced by Llama (j) is " <i>A cover of Ghost of Tsushima features a striking fusion of traditional Japanese warrior attire and futuristic machinery, in a unique and captivating image of a samurai-inspired mecha.</i> "	138

List of Tables

4.1	Account discovery comparison between prior string generation models, and UsernameCrazy’s primitive models, primitive models with self-repetition, and primitive models with model stacking.	49
4.2	User categorization as determined by three independent raters following the official X documentation. The accounts ‘@realdonaldtrump’, ‘@ariana-grande’ and ‘@aamir_khan’ shown in light grey are not included in our experiments.	51
4.3	Top five original accounts with the most active username variants on X. ‘Generated Accounts’, ‘Active Accounts’ and ‘Suspended Accounts’ are abbreviated as ‘ <i>Gen. Acc.</i> ’, ‘ <i>Act. Acc.</i> ’ and ‘ <i>Sus. Acc.</i> ’, respectively. Also, we denote the ‘Generated/Active/Suspended Accounts with an edit distance 1-3’ as ‘ <i>Gen./Act./Susp. Acc. ED 1-3</i> ’.	59
4.4	Original accounts within the first 10 and 100 search recommendation results from a total of 929 requests.	61
4.5	Manual classification of suspended accounts.	65
4.6	Manual classification of the active malicious accounts used in dataset. . . .	66

5.1	Accuracy of VGGFace2 with various thresholds. Setting the threshold value to 0.65, the model has the best accuracy.	71
5.2	Competency of <i>URLCrazy</i> , <i>AppCrazy</i> and <i>UsernameCrazy</i>	76
5.3	Classification performance. (P) shows the positive class.	82
5.4	SQUAD results on non-popular accounts.	85
5.5	Similar impersonation attempts across various Online Social Networks. . .	87
6.1	The number of unique active members that use aliases or real names as usernames in Stable Diffusion, Midjourney, DALL·E and DALL·E 2 subreddits.	94
6.2	Comparison of the size of the DiffusionDB datasets. DiffusionDB Large is a superset of DiffusionDB Small.	108
6.3	Top 1-to-5 accuracy scores on different experiment setups.	117
6.4	Embeddings similarity between the original NLI and the generated image’s caption produced with BLIP, BLIP-2 and LlaVA.	119
6.5	The Top-5 accuracy of the best performing (in parenthesis) Authorship Inference models using NSFW images.	119
6.6	Top-5 Accuracy of Best Performing Output Attack ($\mathcal{A}_o$) models without and with data augmentation in model-dependent (*) and model-independent inference settings.	123
6.7	The accuracy of the best performing (in parenthesis) Dementia Inference models using images from Stable Diffusion v2.1 and Latent Diffusion. . .	128
6.8	Semantic similarity between original and paraphrased NLIs, using PEGASUS, Llama3 and Mistral.	133

Chapter 1

Introduction

Online identities and their associated identifiers—such as usernames and generated digital artifacts—are fundamental to modern digital interactions. However, they are increasingly targeted by adversaries seeking to exploit vulnerabilities for impersonation, privacy violations, and other malicious activities. This thesis investigates the multifaceted challenges involved in safeguarding these online identifiers, with a particular focus on identifying and mitigating threats through effective detection mechanisms and protection strategies.

1.1 Overview

In today’s digital age, online identities play a vital role in personal, professional and societal interaction [78]. The increasing dependence on digital platforms for communication, commerce and social engagement has increased the necessity to ensure the integrity and security of these identities [9]. Yet, this reliance comes with vulnerabilities. Threats like identity theft, impersonation, and privacy breaches undermine the online trust [147] [120]. As a result, the consequences span across personal, economic as well as organizational do-

mains.

This thesis tackles these challenges by exploring the mechanisms, strategies and implications of online identity inference and impersonation. The research focuses on two key areas: (1) username squatting on Online Social Networks, and (2) the privacy and security vulnerabilities introduced by emerging technologies such as Text-to-Image generation models.

In the first area, username squatting, malicious actors exploit subtle variations in usernames to impersonate legitimate accounts, create confusion, or facilitate fraudulent activities [252]. Despite its prevalence, the broader implications and underlying mechanisms of this practice remain insufficiently studied [7]. By developing a new detection tool called UsernameCrazy and by introducing SQUAD, a novel end-to-end framework, this research not only sheds light on the scale and impact of username squatting but also offers practical methods for mitigation.

The second focus area addresses *vulnerabilities in text-to-image models*, such as Stable and Latent Diffusion. While these models offer remarkable creative applications, they also pose risks [46] [376][232], including leaking sensitive information and enabling attribute inference. By defining and rigorously evaluating new adversary models, including a multi-modal one, this thesis underscores the need for stronger defenses and privacy-preserving measures.

Through a combination of empirical research, innovative methodologies and the development of robust frameworks, this thesis advances the understanding of online identity threats and proposes actionable solutions. Beyond identifying the weaknesses of existing systems it paves the way to a safe and more trustworthy digital ecosystem. Consequently, this research represents a significant step forward in protecting online identities in an era of rapidly evolving technologies and societal shifts.

1.2 Thesis outline, state-of-the-art and contributions

1.2.1 Thesis Outline

This section provides an overview of the thesis, introducing the core issues surrounding online identity inference, impersonation, and the strategies for mitigating these threats. It also outlines the key contributions made in addressing these challenges.

The thesis is centered on a) how can users impersonate online as well as what are the novel approaches for malicious actors to impersonate accounts online?, b) what strategies adversaries use to infer the identify of the online users? and c) can we defend against these issues? As a result, the thesis focuses on the vulnerabilities of online identities, particularly as they relate to social platforms and emerging text-to-image (T2I) technologies.

In today's digital age, online identities are pivotal, serving as the backbone of personal and organizational interactions. However, these identities are increasingly at risk from threats such as username squatting, impersonation, and privacy breaches. The first part of the thesis focuses on the problem of username squatting, where malicious actors exploit variations of usernames to impersonate legitimate users or cause confusion. To tackle this, we developed a new tool that generates squatted username variants, an integral part of our novel end-to-end detection framework. This framework is designed to effectively identify and mitigate these risks, and its effectiveness has been validated against real-world data, proving its value in protecting online identities. The second part of the thesis explores the vulnerabilities of T2I models, where issues like authorship inference and privacy leakage are prevalent. We introduce several black-box adversary models that can exploit these vulnerabilities while demonstrating that current state-of-the-art defenses fall short in safeguarding user privacy while retaining utility.

While the first part of the thesis offers practical mitigation strategies, the findings from the text-to-image study emphasize the need for further research to develop more ro-

bust defenses. Overall, this thesis advances our understanding of the threats to online identities, offering practical tools and frameworks for detection and mitigation, and highlighting significant gaps in current protective measures. These contributions are crucial for improving the security and privacy of online identities in an increasingly connected world.

1.2.2 Related Work

This section positions the context of the related work within the more general scientific topics discussed in this thesis. Here, we only briefly refer to the related work that has influenced our work, and we mainly categorise this work into four parts:

1. We are working on identifying impersonation attempts on online social networks and developing a novel framework to efficiently detect suspicious attempts. We also discuss other frameworks that aim to uncover and mitigate the improper use of confusingly similar profiles in social networks. Noteworthy works on impersonation and profile name reuse can be found in Mariconti et al. [213], Goga et al. [108] and Kontaxis et al. [166]. Further background analysis is provided in Sections 2.1 and 2.2 and Sections 3.1 and 3.2.
2. Secondly, we are focusing on squatting methodologies for generating username variants. Since we emphasize on squatting techniques, we are reviewing the state-of-the-art in applying squatting methodologies as in *Domain-squatting* [336, 90], *Sound-squatting* [233], *Email-squatting* [312], and additionally, more sophisticated in *Combo-squatting* [163], *App-squatting* [139] and *Voice-skill-squatting* [170]. More details on squatting techniques and tools can be found in Section 3.3.
3. Thirdly, we are exploring authorship identification on text, and more specifically on prompts. We evaluate the latest advancements of model embeddings to identify whether they can perform inference effectively on prompts. Notable work on

embeddings models can be found in Song and Raghunathan [302] and Song and Shmatikov [303]. Additional details on inference attacks and embedding models can be found in Section 2.4 and Section 2.5, respectively.

4. Moreover, we are working on authorship inference from text-to-image diffusion models. We define and evaluate three new black-box attacks (input based, output based and a multi-modal adversarial model) to achieve state-of-the-art performance in authorship identification and dementia inference. Significant work on authorship and dementia leakage can be found in Song and Raghunathan [302] and Yuan et al. [373], respectively. More related work and discussion on diffusion models and authorship inference can be found in Sections 3.4 and 3.5.
5. Lastly, we rigorously evaluate state-of-the-art defense strategies (e.g., paraphrasing and large language models) against our multi-modal attacks. Remarkable works can be found in Zhang et al. [380], Campos et al. [43], Tripto et al. [320] and others. For further details on authorship obfuscation see Section 3.4.

Again, here we only aim to provide a ‘map’ of the related work in this thesis, and we kindly recommend that the reader focuses on the corresponding sections in every Chapter.

1.2.3 Thesis Contribution

Summarising, this thesis offers original contributions in the following areas:

1. It reveals a new strategy that adversaries use to impersonate online, which is called username squatting. It provides an in-depth analysis of username squatting as a significant vector for identity impersonation on social platforms. Also, it identifies the prevalence, characteristics, and consequences of this phenomenon.
2. It develops a novel strategy and uniquely shows that authorship and other attributes, like Dementia, can be inferred from text, images and both. It is the first to explore

the risks of authorship and attribute leakage in text-to-image diffusion models, providing new black-box adversary models for identity inference.

3. It develops tools to help regulating authorities, model provides and online providers to tackle the aforementioned issues. It proposes a novel end-to-end detection framework to mitigate the username squatting phenomenon and evaluates the effectiveness of current state-of-the-art defenses against authorship and attribute inference.

More specifically, the above topics lead to the following research sub-contributions:

- *New Tool & Techniques.* We propose UsernameCrazy, an efficient tool embodying new techniques for generating squatted versions tailored to OSN usernames rules.
- *Measurement Study.* We performed a comprehensive measurement study on X and found that username squatting is a prevalent and growing problem, it can contribute to online confusion, and is used for malicious activity.
- *Detection Framework.* We design, implement, evaluate and apply SQUAD, a novel end-to-end framework that leverages username squatting as a strong signal to efficiently and effectively discover suspicious squatted accounts.
- *New Inference Models.* We have defined and evaluated new black-box adversary models (Text-based, Image-based, and Multi-Modal) for authorship identification in text-to-image models.
- *Discoveries.* We are the first to demonstrate that authorship identification is possible based on text-to-image model's inputs and outputs. We have also demonstrated that T2I images can leak other attributes and more specifically the presence of Dementia.
- *Mitigation Strategies.* We have discussed and rigorously evaluated state-of-the-art mitigation strategies against the developed black-box attacks.

The thesis comprises seven chapters, including the present introductory chapter and the conclusion. Chapter 2 sets the groundwork that is essential for the thesis. Chapter 3 establishes the foundation for the contribution in the subsequent chapters. It includes the necessary knowledge to familiarise the reader with the topic of this thesis and refers to the state-of-the-art related work from the literature.

Chapters 4, 5 and Chapter 6, which respectively treat the aforementioned topics 1-3, represent the core contributions of this thesis, as detailed next:

Chapter 4 In this chapter, we describe our methodology for measuring *username-squatting* on OSNs and present our results. More specifically, we design a comprehensive measurement study which comprises four concrete measurement questions: **(MQ1)** Is *username squatting* a prevalent problem on X ? If it is, then **(MQ2)** how does it contribute to online confusion; **(MQ3)** what are the username characteristics of the most problematic username variants, and **(MQ4)** what is the specific behavior of the squatted accounts (i.e. what is the *purpose* of the owner of such accounts)? Also, to aid our analysis we build a new tool, called UsernameCrazy, to construct thousands of username variants.

Chapter 5 We present the overall design of our framework and evaluate its key components. Using the insight of the measurement results of Chapter 4 we take the first step to address the *username-squatting* issue and we develop a new end-to-end framework, SQUAD, to detect squatted accounts on X . This work was co-authored by Anastasia Borovykh (Imperial College London) and supervised by Soteris Demetriou (Imperial College London). This thesis' author entirely performed the generation models' design, the implementation of the tool (UsernameCrazy) as well as the end-to-end framework (SQUAD).

Chapter 6 We introduce three new black-box adversary models for authorship identification on text-to-image generation models. We compare the performance of the adversary models against various setups/scenarios (e.g., using different training sets, or

whether there is model dependency) and demonstrate their robustness. Also, we rigorously evaluate state-of-the-art mitigation strategies to mask authorship against our proposed attacks. We also demonstrate that T2I images can leak other attributes, and more specifically, the presence of dementia. This work was co-authored by Marios Charalambides (Imperial College London), Jiani Liu (Imperial College London), Yiyang Guan (Imperial College London), Dominika Woszczyk (Imperial College London), Mansi (Imperial College London), Le Thanh Hai (Imperial College London) and supervised by Soteris Demetriou (Imperial College London). This thesis' author performed the design, implementation and application of all adversary models as well as the rigorous evaluation of the defense mechanisms.

This thesis advances the understanding of the security and privacy challenges associated with username squatting on social platforms and the vulnerabilities inherent in emerging T2I technologies. Through the development of a new tool and a comprehensive detection framework, it provides effective methods to identify, quantify and mitigate the risks associated with username squatting. In the context of T2I models, the thesis introduces new and robust black-box attacks for authorship and dementia inference. Additionally, the research offers critical insights into the limitations of current state-of-the-art defenses against authorship inference and attribute leakage in T2I models. The evaluation demonstrates that existing defenses are inadequate in protecting users, underscoring the need for further advancements in privacy-preserving techniques within this rapidly evolving field.

1.3 Dissemination

The work implemented for this thesis is released under open-source licenses. The underlying theory has been exposed to the scientific community through the following publications and presentations:

Publications. Three main publications derive from this thesis :

- **A. Lepipas**, A. Borovykh, and S. Demetriou, ‘*Username squatting on online social networks: A study on X*’ (Proceedings of the 19th ACM Asia Conference on Computer and Communications Security), 2024. Available Online. This conference paper, presented in Chapters 4 and 5, describes the work on identifying and mitigating the *username squatting* issue on online social networks.
- **A. Lepipas**, M. Charalambides, J. Liu, Y. Gua, D. Woszczyk, Mansi, Le Thanh Hai and S. Demetriou, ‘Leaky Diffusion: Attribute Leakage in Text-Guided Image Generation’ (The 25th Privacy Enhancing Technologies Symposium), 2025. Available Online. This symposium paper, presented in Chapter 6 presents studies of authorship leakage in text-to-image generation models.
- **Mansi**^{*}, **A. Lepipas**^{*} and S. Demetriou, ‘Understanding Dementia Speech Alignment with Diffusion-Based Image Generation’ (26th edition of the Interspeech Conference), 2025. Available Online. This conference paper, presented partially in Chapter 6 demonstrates another attribute inference study than authorship.

^{*} The first two authors contributed equally to this work.

Presentations A number of formal and informal presentations were given at various conferences and meetings. The most relevant in chronological order, are:

- **A. Lepipas**, A. Borovykh, and S. Demetriou, ‘*Username squatting on online social networks: A study on X*’ (Proceedings of the 19th ACM Asia Conference on Computer and Communications Security), 2024. The oral presentation is available at [SLIDES].
- **A. Lepipas**, M. Charalambides, J. Liu, Y. Gua, D. Woszczyk, Mansi, Le Thanh Hai and S. Demetriou, ‘Leaky Diffusion: Attribute Leakage in Text-Guided Image Generation’ (The 25th Privacy Enhancing Technologies Symposium), 2025. The oral presentation is available at [SLIDES].

- **Mansi^{*}**, **A. Lepipas^{*}**, and S. Demetriou, ‘Understanding Dementia Speech Alignment with Diffusion-Based Image Generation’ (26th edition of the Interspeech Conference), 2025. This work presented by a co-author and the oral presentation is available at [SLIDES].

^{*} The first two authors contributed equally to this work.

Software. The research described in this thesis has contributed towards improving the understanding of vulnerabilities in online identifiers across OSNs and T2I models, offering tools to address identity threats and privacy leaks.

- SQUAD Framework.
- Attribute Leakage.
- Dementia Speech.

Chapter 2

Background Work

This Chapter aims to provide background knowledge that is essential for the thesis.

2.1 Online Identity

An online identity is the unique persona that an individual, organization, or entity creates and maintains in online spaces. Researchers have highlighted that such an identity emerges through a combination of personal choices, social contexts and the technological features of online platforms [39, 79, 321]. Online identities typically develop as users participate and engage with various digital platforms, including forums, websites and especially Online Social Networks (OSNs). Furthermore, these identities consist of multiple elements such as usernames, profile information, multimedia content, and activity patterns [385], reflecting both users' preferences and the rules or limitations of each platform [247].

Online Identities exist on a spectrum—from fully authentic, where users share their real names and personal details, to pseudonyms or entirely anonymous forms [60, 156].

They play a crucial role in defining how users interact with digital spaces since they

enable communication and support the formation of communities. As a result, they often affect the perception of trustworthiness and authenticity within these environments. Importantly, online identities are not static; they evolve over time as users' preferences, social networks, and platforms' rules change. Consequently, an online identity is closely tied to the governance mechanisms implemented by Online Social Networks, such as account and username policies, which ensure a balance between user freedom credibility but also the platform's integrity [79, 154].

Integration with Online Social Networks

Online Social Networks are digital platforms that facilitate interactions and connections among users. These networks allow individuals to create profiles, share personal information, communicate with others, and participate in various forms of social activities such as posting updates, sharing multimedia content, joining groups, and forming communities. OSNs have become an integral part of modern communication and have profoundly influenced how people interact and share information online.

OSNs typically operate on a client-server architecture, where users interact with the platform through client applications (e.g., web browsers, mobile apps) that communicate with servers hosting the network's infrastructure and data. The servers store user profiles, content, and social connections, enabling seamless sharing and communication among users. Algorithms are often employed to personalize users' experiences by recommending friends, content, or groups based on their interests and activities within the network [165, 51]. As online social networks continue to evolve, it is essential to understand the underlying mechanisms that govern user interactions and platform usage. Certain regulations and guidelines are put in place to ensure a safe and inclusive environment for all users. Next, we explore a range of account and username policies of OSNs.

Policies Shaping Online Identities in OSNs

Account Policy: Account policies govern the rules and guidelines users must follow when creating and managing their accounts on an OSN platform. These policies often

include requirements for user identity verification, restrictions on behavior and content, privacy settings, and actions taken to prevent activities such as spamming, hate speech or harassment. For example, Facebook [220] requires users to provide their real names to prevent anonymous or fake profiles, as clearly stated in their Community Standards [221]. They also provide clear guidelines on what is considered acceptable content and behavior. Similarly, X [355] (former Twitter) enforces policies that emphasize on impersonation, spam, and abusive behavior, as stated in their rules [365].

Username Policy: Username policies govern the selection and usage of usernames, which serve as unique identifiers for users within the OSN. Platforms impose various restrictions on usernames to prevent impersonation, trademark infringement, or offensive language and content. For example, Instagram [145] prohibits usernames that violate trademarks, impersonate others, or contain hate speech, violence, or explicit content [144]. Additionally, some platforms may allow users to change their usernames periodically, while others have stricter limitations to prevent confusion among followers.

It is important to note that each OSN platform has its own set of account and username policies, which users are required to comply with, in order to maintain a balance between personal expression and platform integrity. These policies are typically outlined in the platform's terms of service or community guidelines and are subject to change at the discretion of the platform operators. By understanding these mechanisms, we can better navigate the evolving digital landscape and its implications for social interactions, privacy, and identity.

2.2 Fake Accounts

The proliferation of fake accounts poses a significant challenge to the integrity and security of online social networks. Fake accounts, also known as impostor or fraudulent accounts, are created with the intent to deceive others by misrepresenting the identity, attributes,

or intentions of the account holder. Whether used for spamming, phishing, spreading misinformation, or engaging in malicious activities, fake accounts undermine trust and credibility within online communities [96].

Fake accounts encompass a spectrum of deceptive profiles created with false or misleading information. These accounts may impersonate real individuals, organizations, or entities, utilizing stolen identities, fictitious personas, or automated bots to conceal their true nature [64]. Fake accounts often exhibit suspicious behaviour patterns, such as rapid friend requests, repetitive messaging, or engagement in coordinated campaigns, which can raise red flags for platform administrators and users alike [339].

From a systems security perspective, fake accounts can enable large-scale security breaches, undermine users' trust and influence campaigns [38]. Adversaries exploit fake accounts to conduct phishing attacks, spread malware and manipulate public opinion [21]. The increasing sophistication of fake account creating techniques, complicates detection and requires advanced countermeasures [257].

To combat the proliferation of fake accounts, OSNs implement a range of measures aimed at detection, prevention, and enforcement. These measures may include:

- **Identity Verification:** OSNs require users to verify their identity through age, address, phone number or other forms of authentication to deter the creation of fake accounts [44]. Moreover, OSNs employ *CAPTCHA* [93] as well as photo-based verification [94] in order to limit the suspected number of users. By linking accounts to verified identities, platforms can enhance accountability and reduce anonymity, thereby discouraging malicious actors from engaging in deceptive behaviour.
- **Manual Reporting and Review:** OSNs empower users to report suspected fake accounts for review by platform administrators. Human moderators may conduct manual investigations to assess the authenticity of reported accounts, examining factors such as profile completeness, activity history, and adherence to community

guidelines [327]. Accounts found to be in violation of platform policies may be suspended, banned, or subjected to additional scrutiny.

- **Automated Detection:** OSNs deploy automated algorithms and machine learning models to detect suspicious account behaviour indicative of fake or fraudulent activity [63]. These systems analyze patterns such as account creation rate, posting frequency, and interaction dynamics to identify and flag potentially fake accounts for further review.
- **Educational Campaigns:** OSNs educate users about the risks associated with fake accounts and provide guidance on identifying and reporting suspicious activity. Educational campaigns may include tips for recognizing fake profiles, safeguarding personal information, and exercising caution when interacting with unfamiliar users or content [350, 8].
- **Law Enforcement:** In cases involving criminal activity or security threats, OSNs collaborate with law enforcement agencies to investigate and mitigate the impact of fake accounts. By sharing information and coordinating efforts with relevant authorities, platforms can enhance their ability to identify and apprehend perpetrators engaged in fraudulent behaviour [351].

Despite these efforts, the battle against fake accounts remains an ongoing challenge for OSNs, necessitating continuous innovation and collaboration across stakeholders. By adopting a multifaceted approach encompassing automated detection, identity verification, user reporting, education, and collaboration with law enforcement, OSNs can mitigate the risks posed by fake accounts and safeguard the integrity of their platforms.

2.3 Harmful Content

As digital platforms continue to expand, the proliferation of harmful content has emerged as a critical concern within online social networks (OSNs). Harmful content encompasses

a broad spectrum of materials that can inflict emotional, psychological, or physical harm upon individuals or communities [105]. The rapid spread and amplification capabilities of OSNs exacerbate the impact of such content, posing serious challenges for users' well-being and safety [244]. From hate speech and cyberbullying to misinformation and graphic imagery, the types of harmful content encountered on OSNs are diverse and multifaceted [238]. Next, we describe several of the most popular types of harmful content appear on OSNs [12, 26].

Types of Harmful Content

- **Hate Speech:** Hate speech includes discriminatory or derogatory language targeting individuals or groups based on factors such as race, ethnicity, religion, gender, or sexual orientation [203]. It fosters division and intolerance within online communities, perpetuating prejudice within online communities.
- **Misinformation & Disinformation:** Misinformation refers to false or misleading information spread unintentionally, whereas disinformation involves the deliberate creation and distribution of deceptive content to manipulate public opinion [345]. Both phenomena pose significant challenges to the integrity of online discourse and undermine trust in information sources.
- **Cyberbullying & Harassment:** Cyberbullying uses digital platforms to harass, intimidate, or humiliate others [168]. It includes behaviours such as online harassment, trolling, and public shaming, which can have detrimental effects on victims' mental health and well-being.
- **Graphic & Violent Content:** This type of content includes images, videos, or text depicting explicit or disturbing material, including but not limited to gore, self-harm, or explicit violence. Exposure to such content can lead to trauma, desensitization, or adverse psychological reactions among viewers [84].
- **AI-Generated Content:** With the advancements in artificial intelligence (AI) and

especially into text-to-image generation models, a new wave of harmful content has emerged [53]. While these technologies can offer creativity and innovation, they also have the potential to produce problematic content, such as violent or manipulated images designed to mislead or harm individuals. This content can also create false narratives, invade privacy, or create exploitative material, necessitating robust moderation frameworks that can keep pace with the latest advancements.

Strategies for Mitigating Harmful Content

To tackle the spread of harmful content, OSNs employ a variety of strategies and mechanisms aimed at content moderation and enforcement of community guidelines [17]. These strategies may include:

- **Human Content Moderation:** OSNs employ teams of human moderators tasked with reviewing reported content and enforcing platform policies [267]. Human moderators provide contextual understanding and nuanced judgment, which is essential for assessing the intent and impact of content more effectively. However, this approach can be resource-intensive and subject to biases or inconsistencies.
- **Automated Content Moderation:** In conjunction with human moderation, OSNs leverage automated tools and algorithms to detect and remove harmful content proactively [282]. These systems employ machine learning algorithms trained to identify patterns associated with harmful content, such as hate speech or graphic imagery. While useful, automated moderation is not without limitations and may sometimes result in false positives or evasion tactics by malicious actors.
- **Feedback and Community Reporting:** OSNs allow users to report and flag content that violates community guidelines or constitutes harmful behaviour [215]. Community reporting mechanisms enable swift identification and removal of offending content, leveraging the collective vigilance of users to maintain a safe and respectful online environment.

- **Accountability and Transparency:** OSNs strive to enhance transparency and accountability in content moderation processes by providing clear guidelines, transparency reports, and avenues for user feedback [164]. Open dialogue helps build trust with users and demonstrate a commitment to effectively fight against harmful content.

Despite these efforts, addressing harmful content remains an ongoing challenge for OSNs, requiring continuous innovation and collaboration across all stakeholders [106]. As the digital landscape evolves, OSNs must remain vigilant in protecting users' well-being and safety while preserving the principles of free expression and open dialogue [29].

2.4 Property Inference Attacks

Property Inference Attacks (PIAs) constitute a significant concern particularly within the domain of privacy preservation and anonymity. These attacks aim to extract general characteristics of a targeted machine learning model's training data, which the owner of the model intends to keep undisclosed. Based on the nature of the targeted properties and the methodologies employed by attackers, PIAs can be classified into several categories.

- **Membership Inference Attacks :** These attacks are designed to determine whether a specific individual's data was included in a trained machine learning model's dataset. The adversary aims to infer if a particular data point was part of the training data, potentially compromising the privacy and confidentiality of individuals' information [297, 275, 277, 218].
- **Attribute Inference Attacks :** These attacks aim to deduce specific attributes or characteristics of individuals, such as age, gender, personal preferences or sexual orientation, based on observable data features available in public or semi-public contexts [265, 47], like social media profiles, user-generated content or online behaviour.

- **Authorship Inference Attacks :** These attacks seek to uncover the true identities of anonymous or pseudonymous users within a given dataset by comparing their linguistic patterns, writing styles as well as behaviour attributes [210]. Adversaries often leverage different techniques, like stylometric analysis, sentence structure, punctuation usage or overall writing style to establish statistical correlation that can reveal an individual's identity. Moreover, with the advancements in machine learning, adversaries have discovered new and more effective inference techniques [271, 293]. Algorithms are now capable of analysing large volumes of text to identify patterns that would be challenging for humans to recognise, increasing the accuracy of identifications.
- **Location Inference Attacks :** These attacks focus on revealing the geographical locations of individuals based on their digital footprint [169], which can include social media activity, global positioning system (GPS) data or other online interactions. The pervasive use of smart devices as well as the integration of location-based services increase the potential for location inference attacks [229].

Formalization of Authorship Inference

Authorship inference, a specific form of property inference, focuses on determining the authorship of a given text or document based on linguistic and stylistic features. We first provide the general definition. Given a set of documents $D = \{d_1, d_2, \dots, d_n\}$ and a set of authors $A = \{a_1, a_2, \dots, a_m\}$, the goal of an Authorship Inference attack is to infer the mapping $f : D \rightarrow A$ such that each document d_i is associated with its author a_j . The attack can be formalized as an optimization problem where the objective is to maximize the likelihood of the correct authorship mapping based on stylistic features (e.g., Stylometry [133]) extracted from the documents. Let $S(d_i)$ represent the stylistic feature vector for document d_i , and $P(a_j|S(d_i))$ represent the probability that author a_j wrote document d_i based on the stylistic features. The optimization problem can be stated as:

$$\max_f \prod_{i=1}^n P(f(d_i)|S(d_i)) \quad (2.1)$$

This formalization assumes that the stylistic features are sufficient to determine authorship and that there is a probabilistic model capable of estimating the likelihood of authorship given these features.

Stylometry

Stylometry [133], a subfield of computational linguistics, plays a pivotal role in authorship inference by analyzing patterns of writing style, syntactic structure, vocabulary usage, and other linguistic attributes [132, 305]. Next, we list several of the most commonly employed stylometry features in authorship inference tasks:

- **Lexical Features:**

- **Word Frequencies :** Distribution of word frequencies and vocabulary richness like the type–token ratio [132].
- **Character N–grams :** Frequencies of character sequences, reflecting stylistic nuances and orthographic patterns [114].
- **Hapax Legomena & Dis Legomena :** Words appearing only once or twice in the text, which can be indicative of author’s unique vocabulary [136, 41].

- **Syntactic Features:**

- **Sentence Length :** Average sentence length and variability, reflecting stylistic preferences and syntactic complexity [159].
- **Syntactic Structure :** Analysis of parse trees as well as dependency relations to uncover complex grammatical patterns [128].

- **Part-of-Speech Tag Frequencies** Distribution of Part-of-Speech (POS) tags, indicating grammatical patterns [305].
- **Punctuation Usage** : Patterns of punctuation marks, such as commas, semi-columns, periods, ellipses, exclamation points, or dashes [176].
- **Semantic Features**:
 - **Topic Modelling** : Distribution over latent topics using techniques like *Latent Dirichlet Allocation* (LDA) to identify thematic patterns [289].
 - **Semantic Dependencies** : Connections between words and their corresponding meanings, including both thematic structures and semantic roles [16].
- **Structural Features**:
 - **Sentence & Paragraph Structure**: Patterns in paragraph organization, such as sentence initiation or use of headings [112].

2.5 Embedding Models

Embedding models are fundamental tools in the machine learning domain. These models serve to translate raw input, typically discrete, into a low-dimensional vector. This vector encapsulates the semantic meaning of the original input data and finds applications in several downstream tasks, like classification, retrieval and clustering [30].

Traditionally, embedding models have been extensively used in natural language processing (NLP) in order to represent i) words, ii) sentences and iii) documents as continuous vectors. Early models such as Word2Vec [223], Glove [251] and FastText [37] focused on learning semantic and syntactic relationships between words by leveraging language's distributional properties.

More advanced models like BERT [74] introduced contextualized embeddings. Such embeddings capture the meaning of the words in context by leveraging the transformer architecture [331]. Variations and extensions of BERT, such as ALBERT [177], RoBERTa [196] and Sentence-BERT [263] have further improved the capabilities of embedding models to represent both sentences and longer text sequences effectively.

While these models have excelled in NLP tasks, there has been a growing interest in embedding models that can handle multiple modalities, like text and images. Particularly, multi-modal embedding models aim to map semantically related data from different modalities into a shared vector space. These new methods can allow for richer and more aligned text and image representations.

A seminal work in this area is CLIP (Contrastive Language-Image Pretraining) [254]. CLIP trains an image and a text encoder together on a significant large dataset of image-text pairs, learning to identify connections between these two modalities. The resulting model embeds images and their corresponding textual descriptions into a shared vector space, where the embeddings' distance indicates their semantic similarity.

Similarly, Vision Transformers (ViT) [81] apply transformer architectures directly to sequences of image patches, achieving outstanding performance in image recognition tasks. Moreover, ViT can be combined with text encoders in order to enable multi-modal representations.

More recently, models like BLIP-2 [188], KOSMOS [142] and PaLM-E [82] have further improved the multi-modal embeddings by integrating large language models (LLMs) with visual understanding. Such models excel at tasks like visual question answering and image captioning.

To the best of our knowledge, we are the first to use representations from such multi-modal models to combine NLIs with T2I images to conduct highly effective attribute inference attacks.

2.6 Summary

We presented, in this Chapter, the fundamental background work for the thesis. We started from defining Online Identities and their integration with Online Social Networks, which we will focus on, as well as explaining OSNs' policies against fake accounts and harmful content. Then, embedding models were highlighted for their role in creating semantic representations for single and multi-modal tasks, supporting various generation applications. Additionally, we reviewed a typical operational mechanism for authorship attribution before closing with the background about the architectures of the Generative Adversarial Networks and Diffusion Models.

In the next Chapter, we will explain in more detail the current state-of-the-art research works related to the topic areas of this PhD thesis. Moreover, we will deliver critical analyses of these works and summarize the research gaps we particularly tackle in this thesis.

Chapter 3

Related Work

This Chapter establishes and critically analyzes the current state-of-the-art research works which are relevant to this thesis. Within this Chapter, we also emphasize where our contributions are specifically as well as the research gaps they address to highlight the significance of our work.

3.1 Account Threats on Social Networks

Sybil Attacks. Prior work has demonstrated the successes of automatically creating cloned profiles in social networks without proposing a methodology to detect them [34] and has studied the prevalence of profile name reuse on OSNs [213, 212]. To tackle such issues, researchers have presented various ways of detecting Sybil attacks on OSNs using e.g., visual profile similarity [372], string matches and users’ relationship measurements to identify whether one’s account has been a victim of identity theft [166], the network relationship and attribute similarity [150], honest networks to reveal dishonest nodes via an inference engine [70] and graph properties to rank profiles according to their perceived

likelihood of being fake [44]. Lastly, Mohaizen and Kim [227] presented a comprehensive comparison between proposed Sybil defenses until 2011. SQUAD instead operates by combining username squatting techniques and static profile features to perform malicious classification.

Impersonation Attempts. Zarei et al. [374] focus on identifying political bots and impersonators targeting popular accounts on Instagram [145]. By using a small dataset of 3 politicians and crawling the reactions on their public posts they try to discover profiles with similar characteristics resulting with 167 impersonation attempts. Similarly, [375] focuses on identifying the content generated by impersonators in Instagram by using a dataset of 15 celebrities. To identify impersonators they collect posts containing associated hashtags with the targets and analyse their behaviour; their proposed system reaches 86% accuracy. More relevant to our work is the Doppelganger bot attack study [108], which detects whether a pair of accounts that portrays the same person is an impersonation attack (victim-impersonator pair) or whether one identity is aware of the other (avatar-avatar pair) and thus be owned by the same user. However, Goga et al. work [108] requires manual selection and analysis of similar profiles and relies on expensive social graph traversals to identify potential impersonation attempts of non-celebrity accounts. Such techniques are only practical when the target accounts are within a small distance (in the close neighbourhood) from the account they are impersonating, which might not necessarily be the case with all username variants. The authors find only 3 impersonation attempts against celebrities out of the 166 potential squatted pairs that have manually identified and collected within a three month period. In contrast, we show that popular accounts are targets of impersonation and confusion attacks. A more detailed discussion considering the main reasons behind our conflicting findings can be found in Section 4.4.1. Notably, we are the first to highlight that username squatting is a central strategy to such attacks and use it to design an efficient detection tool. Furthermore, according to prior definitions an avatar-avatar pair is not considered dangerous. We find that most of our malicious cases (see Table 4.6 in Section 4.4.4) belong to the avatar-avatar pair category

(521/540 accounts) making these accounts an overlooked problem. Finally, our framework does not rely on the following assumptions: a) an impersonator should have any kind of relationship with its target (e.g., being friends, commenting or reacting on tweets) and b) in the close neighbourhood of an impersonator we find more impersonators. Instead, with our framework an impersonator can be discovered while being anywhere in the social graph.

Bot Accounts. Previous studies on identifying and profiling bots have focused largely on malicious bots and more precisely on the behaviour patterns of accounts, concluding that they confuse users and pose social, financial and political hazards [14, 48, 287, 377, 101]. Oentaryo et al. [237] focuses on benign bots; tools also exist for profiling malicious bots [281] and detecting potential misinformation spreading [143]. In our work we use Botometer [281] to classify existing squatted accounts and gain a better understanding of their behaviour. However, Botometer uses the characteristics of bots [281], and while *botness* is a good signal it is neither necessary nor sufficient for detecting the more general confusing accounts as well as impersonations. More specifically, according to X [324], Botometer is far from perfect because it uses a limited approach to identify bots; it ignores several use cases, like an account that is extremely active and posts tweets for a specific topic does not necessarily mean that is a bot or spammer. The aforementioned works deviate from the content of this thesis, which focuses on impersonation attempts while treating any bot only as a *suspicious* account, rendering direct comparisons inappropriate.

Spam Accounts. Earlier works [386, 369] study a different problem, that of *spam* accounts on OSNs. More specifically, [386] creates passive honey-profiles, logs the friend requests and messages these profiles receive and tries to identify anomalous behaviour on these activities. Yang et al. [369] use honeypots and blacklists to identify spammers, extract common evasion tactics based on their behaviour, evaluate previous proposed detection techniques [178, 307, 338] and develop a new system to identify spammers in Twitter. Similarly, others [113, 179, 317] focus solely on developing suspicious URL detection systems for Twitter and other web services. Xu et al. [368] propose a heavyweight

ML framework which leverages features based on the social graph and the direct and indirect neighbour properties to detect users who violate the policies of an OSN, including bots, spammers and fake accounts. In contrast, the features used in SQUAD were chosen to be easy and computationally cheap to collect and were based on analyzing specifically impersonation accounts.

3.2 Fake Content on Social Networks

Fake content—ranging from misinformation, disinformation to other deliberately deceptive materials—is content designed to misrepresent reality aiming to influence peoples’ perception, beliefs or behaviours. Within digital platforms, the generation and spread of such content often depend on manipulated or bogus online identities—such as bots, sockpuppets and impersonators—that leverage crafted profiles and social connections in to appear legitimate and trustworthy [80, 171, 97, 55].

The origins of fake content can be traced back to the early days of the Internet, but the advent of OSNs has magnified the scale and impact of these phenomena. Classic studies, such as Zollmann et al.’s seminal work [390], underscore the early recognition of automated entities contributing to the dissemination of misleading information. Additionally, the groundbreaking analysis by Friggeri et al. [100] illuminates the dynamics of rumour propagation in online environments, laying the foundation for subsequent research in understanding the virality of fake content. A comprehensive examination of misinformation on Twitter, as explored by Vosoughi et al. [334], elucidates the intricate patterns of information diffusion and the alarming prevalence of false information. Furthermore, the study by Shu et al. [299] provides insights into the application of data mining techniques for identifying and combating misinformation.

Recent advancements in technology and research methodologies have paved the way for innovative approaches to combat the proliferation of fake content and news. Notable

contributions include the utilization of machine learning algorithms to detect deception by examining linguistic patterns, as demonstrated by Wang [341]. In parallel, Kumar and Shah [172], Subrahmanian et al. [308] and Varol et al. [330] have explored network-based and behavioural analysis methods to detect coordinated identity manipulation as well as the deceptive or malicious content they disseminate, offering new solutions to tackle complex challenges.

Ultimately, fake content and online identities are closely related. By exploiting the anonymity and widespread reach of OSNs, deceptive actors craft believable online identities, often gain peoples' trust and inject false stories into everyday online discussions [171, 172].

3.3 Squatting Techniques and Tools

Squatting issues have been studied in the past within different domains [313, 233, 163, 139, 170, 68, 381, 312] and have been examined on how they can deceive users and lead them to become victims of illegal activities (e.g., identity and money theft by selling goods or to download and install malevolent applications). One of the most popular squatting attacks is the *domain squatting* attack. An attacker attempts to register a domain name that is close to an active one to steal traffic that was intended for the legitimate domain name. Currently, the most advanced tool for domain squatting is *URLCrazy* [336]. In light of this, some of the most widely-used squatting techniques are:

- *Typo-squatting*, which exploits simple typographical mistakes [313, 11].
- *Sound-squatting*, which leverages homophones [233].
- *Combo-squatting*, which attempts to combine key words like 'verification' or 'payment' with popular trademarks [163, 139].

- *Voice-skill-squatting*, which uses homophones to confuse virtual assistants and make them call malicious skills [170, 381].
- *Email-squatting*, which also manipulates typographical errors [312].
- *Bit-squatting*, which exploits bit-flip errors in DNS requests [68].

We note that *Combo-squatting* in contrast to *Typo-squatting* lacks a generative model. An adversary can postfix and/or prefix any word bounded only by the maximum number of allowed characters [363], resulting in an unmanageable number of usernames. Other works, like [312], have tried to measure whether *email typosquatting* is a common phenomenon by registering a number of typosquatting popular domains to identify how often users make typo mistakes when sending emails. Hu et al. [139] proposed a new tool, *AppCrazy*, which leverages several squatting models to generate variations of app identifiers. They conducted the first in-depth measurement study proving that *app squatting* is a prevalent problem in the mobile ecosystem. However, these works are limited in their ability to generate compatible OSN usernames as they suffer from several limitations (see Section 5.4.1). To the best of our knowledge, we are the first to a) develop a string generation tool compatible with OSN usernames, b) apply squatting methods to the discovery and characterization of squatted usernames on OSNs and c) study how adversaries can adapt them to provoke information confusion to OSNs.

3.4 Authorship Inference & Obfuscation

Authorship Inference. Authorship inference has traditionally relied on stylometric features such as character n-grams [279, 278, 305, 286], syntax parse trees [384], words, part-of-speech tags [140], and topic information [387] incorporated into CNN models [298]. While effective, these methods require extensive feature engineering and are limited in their scalability for large datasets.

Other works leverage transformer-based architectures such as BERT, RoBERTa, and BERTAA [74, 91, 149, 196] which are fine-tuned to automatically capture writing style features from raw text [75]. Song and Ragunathan [302] leveraged embeddings to train classifiers to infer authorship in Book corpus. These methods reduce the need for explicit feature engineering, making them more scalable and adaptable to various datasets.

However, none of these approaches have explored authorship leakage in T2I models. To address these limitations we uniquely leverage data augmentation and multi-modal text-image alignment models to achieve SOTA performance with a large number of authors, limited training sample sizes, and in classifier-dependent and independent ways.

Inferences in T2I Models. Previous work [290] illustrated how to attribute a generated image to its source T2I model, while Carlini et al. [46] demonstrated how to extract training examples from diffusion models. Also, Qu et al. [253] showed how prone these models are to generate unsafe or undesirable content, raising concerns about their robustness. Furthermore, Huang et al. [141] employ GPT-4 Turbo with Linguistically Informed Prompting (LIP) to discriminate among 20 candidate authors in the Blog corpus, but their macro-F1 drops from ≈ 0.79 with 10 authors to ≈ 0.55 with 20 authors, indicating diminishing returns as the author set grows. However, our threat model is substantially larger—200 authors—where the expanded combinatorial space would likely further degrade performance and increase inference cost. Consequently, we do not employ an LLM-based authorship inference for our input-side attack. In contrast, we preliminary report here that using embedding models at our scale of 200 authors we achieve a 68.6% accuracy (see Table 6.3 in Section 6.4.1). To the best of our knowledge, no study has been conducted on inferring the author of T2I NLIs and images. This gap highlights the need for a comprehensive investigation of the risks associated with authorship leakage in T2I models.

Authorship Obfuscation. The obfuscation of authorship in text has been extensively researched to enhance privacy and minimize the risk of attribution. Some works follow

a style transfer approach [208, 294] to mask authorship by modifying linguistic features, while others use rule-based modifications [33, 256, 115, 366] to systematically adjust linguistic features, such as replacing words with synonyms or restructuring sentences. Text summarisation [235] offers an alternative in authorship obfuscation by reducing the granularity of the input text, effectively diluting stylistics markers. While such an approach lowers the risk of attribution, it also leads to a significant semantic loss, limiting its usability in contexts requiring high accuracy outputs.

More recently, the use of LLMs for paraphrasing has been a promising direction. Several techniques [43, 99, 205, 320] leverage the generated capabilities of GPT and PEGASUS [380] to produce semantically equivalent yet stylistically distinct outputs. Moreover, obfuscating techniques for machine generated authorship, such as those proposed by Wang et al. [344] and Macko et al. [204] show potential in multilingual contexts but often fall short in preserving utility on downstream tasks. Differentially private methods [95, 216] offer theoretical privacy guarantees, but often compromise utility, especially in T2I pipelines. In addition, UPTON [343] employs data poisoning to prevent authorship leakage. This approach focuses on generating poisoned examples that disrupt the training data of authorship attribution (AA) models. This makes it difficult for the models to accurately attribute authorship by corrupting the feature space with inconsistent authorship fingerprints.

We have selected several representative methods to evaluate the robustness of our attacks including the differentially private approach from Mattern et al [216], and widely adopted open-source LLMs such as Llama [2] and Mistral [3]. A common limitation of those approaches is the negative effect on utility. To gauge this trade-off, we also incorporated the Transformer-based paraphrasing model Pegasus [380] which achieves good semantic preservation. Nonetheless, we have shown in our evaluation, any negative semantic preservation effects in the text input can be compounded in T2I models to further damage the end-to-end utility and highlighted the impending need for more practical obfuscation models to defend T2I users' privacy.

3.5 Text-to-Image Diffusion Models

Diffusion models, also known as diffusion probabilistic models, are a class of generative models introduced by Ho et al. [129]. These models have gained significant attention in recent years for their ability to generate highly realistic data, particularly images, through a unique iterative process. Similarly, Generative Adversarial Networks (GANs) [110] have long dominated image synthesis; early text-to-image (T2I) methods relied heavily on GAN architectures [261] [379]. However, diffusion models embrace a fundamentally different paradigm by explicitly modelling the data’s probability distribution. The model generates samples by gradually transforming a simple initial distribution, like Gaussian noise, into a complex, structured representation that closely resembles real data. Unlike GANs, which involve a delicate balance between a generator and a discriminator, diffusion models are trained in a single-stage process to capture the underlying distribution of the data more directly. Below, we delve into the architecture, objective function, and training procedure that enable diffusion models to achieve these impressive results.

Architecture:

At the heart of a diffusion model is a sequence of diffusion steps that progressively transform an initial distribution, often Gaussian noise, into a distribution resembling real data. Each step in this process applies a small, learned perturbation, which slightly refines the noisy input in the direction of the target data. Over many steps, the noise transitions from an unstructured state to one that captures the rich features and structures present in the data distribution. This gradual transformation process can be visualized as a trajectory in data space, where each step takes a tiny step closer to the desired data distribution.

Diffusion models often leverage neural networks, such as U-Nets or transformers, to parameterize these transformations. The neural network learns to denoise the data at each stage, resulting in smooth but also high-quality outputs. Unlike GANs, which learn to generate an entire data sample in a single step, diffusion models add small details over

many steps, allowing for fine-grained control over the final output.

Objective Function:

The training of diffusion models involves maximising the likelihood of observing the real data under the generated samples. The objective function for training diffusion models is typically based on maximizing the log-likelihood of the data given the model parameters. Specifically, given a set of observed data samples $x_1, x_2, x_3, \dots, x_n$ the objective is to maximise the log-likelihood:

$$\max_{\theta} \sum_{i=1}^n \log p_{\theta}(x_i) \quad (3.1)$$

where $p_{\theta}(x_i)$ represents the probability density function of the diffusion model with parameters θ .

Training Procedure:

- **Diffusion Process:** During training, the diffusion models apply a series of diffusion steps to the input noise vector to generate samples. Each diffusion step applies a learned transformation to the input noise, gradually blurring it until it resembles the target data distribution.
- **Parameter Estimation:** The parameters of the diffusion models are learned by maximizing the log-likelihood of the observed data. This typically involves using Stochastic Gradient Descent (SGD) [160] or variants such as Adam [162] to optimize the objective function.
- **Generation:** Once trained, the diffusion models can generate new samples by iteratively applying the learned diffusion process to random noise vectors sampled from the base distribution.

Hence, this research directs its attention on Stable Diffusion [268], a widely recognized and publicly available text-to-image model.

Latent Diffusion. Latent Diffusion (LD) [269] serves as a diffusion model in text-to-image generation. Latent Diffusion performs diffusion in a compressed latent space rather than in the pixel space. This approach reduces computational requirements while maintaining high-quality image generation. Latent Diffusion employs an autoencoder to map images to a lower-dimensional latent space, where the diffusion process is applied. After diffusion, the decoder reconstructs the image from the latent representation. The utilized model has been pre-trained on LAION-400M [284], a subset extracted from the larger LAION-5B dataset [285]. Additionally, Latent Diffusion incorporates the text encoder of CLIP [241] to direct the creation of synthetic images.

Stable Diffusion. Stable Diffusion (SD) [268] is a latent diffusion model for various tasks, like text-to-image generation, image-to-image translations and inpainting/outpainting. Stable Diffusion leverages the efficiency of Latent Diffusion to enable high-resolution image synthesis with feasible computational resources. The model we explore is Stable Diffusion Version 1 which was pretrained on 256×256 images of LAION2B-en[367] and then finetuned on 512×512 images obtained from a subset of LAION-5B [285]. CLIP ViT-L/14 [241] is used as a text encoder for the diffusion model.

DALL-E, DALL-E 2 and DALL-E 3. DALL-E [259] introduced a transformer-based approach[331] to text-to-image generation, modelling the joint distribution over text and image tokens, developed by OpenAI [242]. The model is trained to autoregressively predict image tokens given text tokens, enabling the generation of novel images from textual descriptions.

DALL-E 2 [258], the next generations of DALL-E, enables better semantic understanding and higher image fidelity. The model uses a two-stage process: i) it first maps text to image embeddings using a diffusion prior and ii) it decodes the aforementioned embeddings into images by leveraging a diffusion decoder.

DALL-E 3 [239] is among the most renowned models for text-to-image generation. The biggest advancement is that it integrates more sophisticated language understanding and image generation techniques. Particularly, DALL-E 3 leverages the capabilities of GPT-4 [240] allowing the model to interpret and generate images from complex and nuanced textual descriptions with greater quality.

Lastly, all DALL-E models were pre-trained on a subset of LAION-5B dataset [285] and use different versions of CLIP [254] embeddings.

Imagen. Imagen [276] is a diffusion-based text-to-image model that emphasizes the importance of large pre-trained language models in understanding complex textual descriptions. The model utilizes the T5 [255] text encoder to process inputs and demonstrates that scaling the language model component significantly enhances image generation quality. Imagen achieves state-of-the-art results in terms of both alignment with textual input and image quality.

Midjourney. Midjourney [135] is a commercial text-to-image model known for producing artistic and aesthetically pleasing images. While detailed technical information is not disclosed, it is believed to utilize diffusion or GAN-based architectures optimized for stylistic image generation.

In our work, we utilize Stable Diffusion (Versions 1 and 2.1) and Latent Diffusion [269], two widely recognized and publicly available text-guided image generation models.

3.6 Image Captioning

Several studies [310, 301, 131] have used CNNs and RNNs to reconstruct descriptions from images. Closest to our work, Croitoru et al. [65] tries to recover the embeddings of the NLIs used to generate the images but without identifying their authors. In contrast, our A_o attack models leverage only the generated images to identify their NLI authors.

A number of works [127, 118, 201] have applied the encoder-decoder architecture while others [185, 383] have been inspired by the BERT model [74].

Other Transformer-based image-text alignment models such as BLIP [189] and CLIP [254] can be leveraged for image captioning. Similarly, U-Net [197] is a convolutional neural network (CNN) that operates in the latent space where the diffusion process occurs. Both CLIP and U-Net have already been employed in diffusion models [259, 258, 268], showing remarkable zero-shot capabilities. Consequently, image captioning seems like a reasonable approach for inferring authorship because if we can accurately reconstruct NLI from T2I images we could potentially infer authorship from captions. In our research we found that this is not possible with the current SOTA captioning models which yield poor semantic similarity with the original NLIs (see Section 6.4). Instead, our A_o and $\mathcal{A}_{i_o}$ models classify authors from transformer-based image embeddings, and they achieve high Top-5 accuracy due to the good alignment between NLIs and T2I images in that embedding space.

3.7 Dementia Classification

The field of dementia classification from speech [202] has grown significantly, particularly since the 2020 *ADReSS* challenge. These models can be categorized into three main types: those using exclusively acoustic data [54, 207], those solely utilizing speech transcriptions [207, 226, 25, 119], and hybrid models combining both [373, 389, 124, 214]. More related to our dementia inference classifiers are models trained on transcriptions. Among those the best performance was achieved by BERT-based models [25]. More similar to our output and input-output dementia inference attacks are the models reported in a preliminary pre-print article by Zhu et al. [388] which combine information from the original “Cookie Theft Picture” with transcriptions to achieve 89.6% accuracy on the *ADReSS* dataset. Similarly with Zhu et al. [388] we also utilize text and image information in our Input-Output inference. However, both our goal and settings are different. Our goal is to study leakage in diffusion models and not to improve dementia classification

and therefore we focus on the ability to learn from joint text–image representations. Also in our setting, we do not assume prior knowledge of the image the participants are trying to describe and focus only on representation extracted from generated images through diffusion models.

3.8 Summary

In this Chapter, we reviewed the essential scientific related work for this thesis. Sections 3.1, 3.2 and 3.3 examined account threats, fake content and squatting techniques, respectively, emphasizing the need for improved detection frameworks. Section 3.4 we looked at authorship inference tools—including stylometric features and transformer-based architectures—as well as state-of-the-art obfuscation techniques such as style transfer, rule-based modifications and LLM-based paraphrasing. Lastly, recent advancements in text-to-image and image captioning models were also discussed in Sections 3.5 and 3.6. These insights setting the stage for the proposed methodologies and experiments in Chapters 4, 5 and 6.

Chapter 4

Measuring Username Squatting on X

Users in online social networks (OSNs) are distinguished through unique identifiers or *usernames*. These identifiers are used for searching for and interacting with other users. Anecdotal evidence suggest that similar usernames might lead to confusion on the network. To make things worse, adversaries—who have been known to opportunistically hijack popular OSN accounts [213]—can deliberately weaponize this feature to impersonate popular or influential users to further their malicious campaigns and advertise content, products or services, spread propaganda, hate content or fake news, or harm influential individuals [292, 314].

Adversaries have targeted unique identifiers in the past using a range of squatting techniques with great success. For instance, traditional typo-squatting attacks target users who mistype a URL [336], mobile-app squatting attacks create mobile apps with names similar to popular apps to increase their installation counts and otherwise fool mobile users [139], and voice squatting attacks target phonetic similarities between the invocation names of voice assistant applications to hijack their communication with their users [170].

However, there is no systematic measurement of similar phenomena on OSNs. Prior works focused on analysing and detecting fake accounts, content and bots on OSNs. These fail to study the extent of squatting on OSNs and mostly rely on expensive graph traversals to identify Sybil nodes [108] resulting in non practical detection methodologies. The rise of disinformation and fake accounts on OSNs already forced regulators to employ stricter policies—such as the *Code of Practice on Disinformation* [87] backed by the European Union’s *Digital Services Act* [88]—to hold OSNs accountable for disinformation and fake accounts on their platforms. Therefore, it is timely and paramount to better understand squatting on OSNs and study techniques for tackling this problem.

This Chapter aims to define the concept of username squatting and provide a comprehensive understanding of its implications within online social networks. It introduces classifications of squatted accounts based on their intent and behaviour, distinguishing between benign, confusion and impersonation accounts. Additionally, it presents the methodology for generating squatted usernames using a custom-developed tool, UsernameCrazy, and discusses the ethical considerations associated with data collection.

4.1 Problem Statement

What is username squatting? OSNs allow their users to create accounts with ease. Users can select any available string for their username¹, which can be used to uniquely identify them across the network. Other users can leverage those usernames to search, mention and interact with account owners. However, this poses a significant threat, as a malicious actor can easily select identifiers at account creation time which are confusingly similar to targeted entities or individuals for malicious purposes. Such techniques have been applied in the past to impersonate web domains (URL squatting [336, 313]), mobile apps (mobile app squatting [139]) and even voice personal assistant skills (skill

¹There is no mention that username policies, in contrast with suspension policies [322, 356], are not the same globally.

squatting [170, 381]) with very high success rates. Similarly, a malicious actor could select a username which is similar with usernames of popular accounts in an attempt to impersonate the target account and/or spread confusion. We call this adversarial strategy *username squatting*. We will refer to squatted usernames as account usernames that are confusingly similar to popular account usernames (such as the ones of verified accounts). Similarly, we will refer to accounts that use squatted usernames as squatted accounts.

Scope. This study defines *username squatting* through *orthographic proximity*—small edit-distance or visually confusable variants of a target handle (often called **typosquatting**). Focusing on typosquatting offers two practical benefits: (i) these variants can be enumerated exhaustively and detected at scale with well-studied string-similarity metrics, enabling the large-scale measurement in Section 4.2; and (ii) prior work shows that even minimal visual perturbations suffice to mislead users and automated systems [336] [139] [312].

We note that this choice narrows the threat surface we quantify. Attackers can also craft *semantically associated* usernames—e.g., *@Bulls23* combined with a Michael Jordan profile (an iconic player of the Chicago Bulls team) photo—to exploit users’ background knowledge without relying on close orthographic resemblance. Such attacks are well-documented in other namespaces under the name ‘combosquatting’ [163] and manifest on social platforms as broader ‘brand-impersonation’ campaigns [7] [375]. Detecting them requires fusing textual, visual, and graph signals—an open problem we leave for future work (see Section 7.1.1). Consequently, the prevalence figures we report should be interpreted as a *lower bound* on the overall impersonation problem; extending our pipeline with semantic similarity models or knowledge-graph embeddings is promising future work.

Definitions. We distinguish benign and suspicious accounts with the latter split into confusion and impersonation.

Benign. Benign accounts have an account name which is a username variant of an original account. However, these users either do not share a profile name, or their profile image

is not similar to the original account, or they explicitly mention that they are a fan or parody users which by X 's policy is allowed [365].

Confusion. We define *confusion* accounts on OSNs to be squatted accounts which share a sufficient number of features with the original accounts and are suspicious (i.e., clearly not benign). Bot accounts are a subset of this class. Confusion can be exploited for malicious purposes (e.g., share links, fake news) and their impact can be amplified through features of OSNs (e.g., the search engine, likes, mentions). Figure 4.1 illustrates a scenario where a verified account on X (identified by the blue check on its name [360]) of a prominent news network mistakenly mentions a fake confusion account “cnnnbrk”, which is *liked* and *retweeted* by numerous users.



Figure 4.1: Typo-mention: *CNN* mistakenly mentions *@cnnnbrk*, a squatted user of the official *@cnnbrk* profile.

Impersonation. We define accounts to be *impersonation* attempts if these have the same features as the original accounts *and* impersonate their *behaviour* either through the tweets or information contained in the bio (e.g., fake verification blue check [157]). Impersonations are a subset of the broader confusion class. While we focus on discovering the more general confusion accounts, we do find a number of impersonations as well (see Figures 4.2a, 4.2b, 4.2c and 4.2d for examples).



Figure 4.2: Impersonation Attempts.

4.2 Measurement Study Design

4.2.1 Ethics

Ethics in Data Collection. We analyzed data available through *X*. We did not extract any personal opinions or other viewpoints linked to individuals as this could potentially be sensitive. In detail, a) all our collected raw data (profile data, tweets with mentions etc.) were captured using official APIs and b) all the unnecessary data that APIs return are discarded immediately. We also c) never share any information with 3rd parties and d) encrypt and keep in a local disk all the data after each experiment. We furthermore follow recommendations in [349] and mask the account names of individuals and do not include any tweets of individual accounts, including only public tweets from the top 97 most popular users.

4.2.2 Data Collection.

Data Collection Methods. We focus our analysis on *X* due to its popularity. *X* has more than 353 million monthly active users in 2023 [236]. We select the 100 most popular accounts of *X* [243]. We call this dataset the ‘*Initial Seed*’. Out of those accounts, at the data collection time ‘@realdonaldtrump’ had been suspended while ‘@aamir_khan’ and ‘@arianagrande’ had been deleted or deactivated. We exclude these accounts from our dataset and all measurements, reducing the total number of accounts from 100 to 97. This is because *X* API does not return information about suspended, deactivated and deleted accounts. This would pose a threat to validity in our methodology [358] as for the following search recommendation and amplification of confusion analysis the results would not include those related to these accounts. However, for every existing user we were able to fetch data like the ‘profile photo, profile name, background photo, bio, location, creation date, followers, followees and tweets’. Moreover, a challenge we had to overcome in crawling *X* stems from API request rate limits which the platform imposes to prevent denial of service (DoS). One could alternatively web crawl the platform using multiple registered accounts. Such an approach, has ethical concerns since it imposes a strain on the studied network while it is also less predictable and reproducible since the client side structure of the network might change at any point. In our study, we leverage *X*’s academic research product² track authorization [359] (2021). This allows us to freely access the full-archive search endpoint of *X* API (with access to public conversations and tweets posted back to 2006) with less strict rate limits [361].

4.2.3 Username Generation (UsernameCrazy).

Username Variant Generation. Our work focuses on studying usernames that might be confusingly similar with ones of already popular accounts. Prior works have lever-

²Academic Research Access was deprecated on March 2023 [354] but the full-archive search is still available through a paid *pro* or *enterprise* level account.

aged seeded network graph traversals for identifying potentially fake or impersonation accounts [108, 368]. Such techniques are only practical when the target accounts are within a small distance from the account they are impersonating, which might not necessarily be the case with all username variants. In contrast, squatting techniques lend themselves well to our problem. Existing tools such as URLCrazy [336] and AppCrazy [139] even though not complete, they have been proven successful in generating complex string variations. Nonetheless, their string generation models are tailored to their specific problems, that of squatting URLs and mobile app identifiers (package names), respectively. As a result, these tools suffer from several limitations because they are restricted to their domains (we refer to them as *prior models*). For example, they produce a large number of strings which are incompatible usernames due to the restrictions that OSNs set for valid usernames. Besides, neither of them supports useful generation patterns, like inserting digits at the beginning or end of a word which would be an easy way for an adversary to automate username variant creations. For clarity we defer a more concrete comparison with prior tools to Section 5.4.1.

To address the limitations of existing tools, we develop an end-to-end tool called **UsernameCrazy** to drive our experiments. UsernameCrazy takes as input a set of account usernames (the *Initial Seed*). For each account, it employs 10 string generation models (we refer to them as *primitive models*) to produce valid username variants. For each variant, the generation model runs until the username reaches the maximum number of allowed characters [363]. The produced variants (*Generated Seed*) are used in our measurement studies. UsernameCrazy generation models are categorized as follows a) Insertion models, b) Deletion models, and c) Misspellings. Figure 4.3 illustrates this taxonomy with examples. Next, we describe each model separately.

- **Vowel Insertion:** Inserts an extra and same vowel character when it finds one, e.g., '@AxlRose' into '@AaxlRose'.
- **Double Character Insertion:** After finding two consecutive and identical charac-

ters, it inserts the same character next to them, e.g., ‘@CNNbrk’ into ‘@CNNNbrk’.

- **Number Insertion:** Adds a number (the same or a different one) both in the end and at the beginning of a username, ‘@Cristiano’ into ‘@Cristiano21’ or ‘@9Cristiano’.
- **Underscore Insertion:** Inserts an underscore both in the end and at the beginning of a username, e.g., ‘@NBA’ into ‘@NBA_’ and ‘@_NBA’.
- **Vowel Deletion:** Deletes a vowel character when it finds one, e.g., ‘@BarackObama’ into ‘@BrackObama’.
- **Double Character Deletion:** Deletes two consecutive and identical characters, e.g., ‘@Twitter’ into ‘@Twier’.
- **Number Deletion:** Deletes all the numbers of the username, one each time, starting both from the end and the beginning of a username, e.g., ‘@AndresIniesta8’ into ‘@AndresIniesta’.
- **Underscore Deletion:** Deletes an underscore in the username when it finds one, e.g., ‘@Ricky_Martin’ into ‘@RickyMartin’.
- **Vowel Character Substitution:** Replaces a vowel with one of the rest vowel characters each time, e.g., ‘@BarackObama’ into ‘@BerackObama’.
- **Common Misspellings & Homoglyphs:** Specific characters in the username are being replaced according to various recognized misspelling patterns [36], e.g., ‘@BarackObama’ into ‘@BarakObama’. In this experiment, we used a small number of common mistakes to remain time-efficient.

UsernameCrazy introduces several important enhancements to improve coverage of username variants compared to existing tools [336, 139]. Firstly, it employs four new generation models, important to username squatting, namely *number insertion*, *number deletion*, *underscore insertion* and *underscore deletion* (see the *blue-shaded* boxes in Figure 4.3).

Secondly, each model is applied repeatedly (*model self-repetition*) on a username. For example, this allows UsernameCrazy’s *Double Character Insertion* model to uniquely produce ‘@Jimmmyfallon’. Thirdly, it supports *model stacking*: after applying a model on a username it takes all the generated variants and applies a second model on them. For example, the combination of *Vowel Insertion* and *Vowel Character Substitution* uniquely generates ‘@BearackObama’, ‘@BoarackObama’ etc. We note that i) during *model stacking* the models are *self-repeated* and ii) the generated usernames of the *primitive models* are not reproduced when *model self-repetition* is applied. We measured the contribution of each of the new features of UsernameCrazy to account discovery and compare it with what can be achieved with *prior models* [336, 139]. The results are presented in Table 4.1. In Section 5.4.1 we perform a more detailed analysis for each of UsernameCrazy’s models.

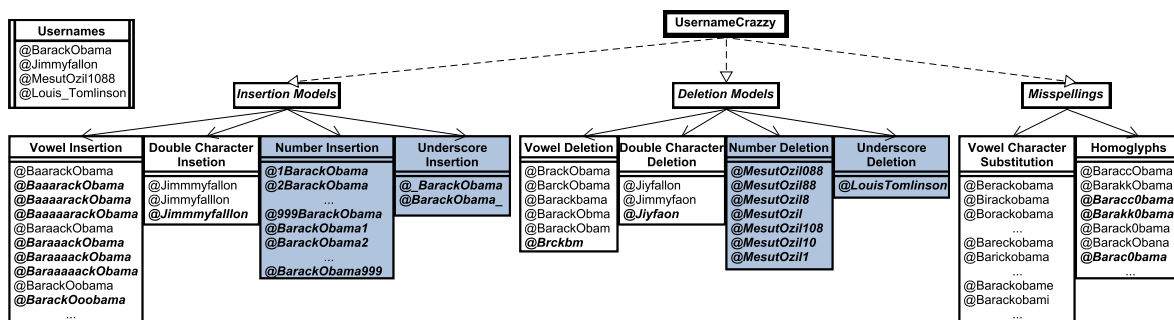


Figure 4.3: Taxonomy of UsernameCrazy’s generation models. The *blue-shaded* boxes indicate models which are not supported by existing tools. In *boldface* are username examples which cannot be generated by existing tools.

Table 4.1: Account discovery comparison between prior string generation models, and UsernameCrazy’s primitive models, primitive models with self-repetition, and primitive models with model stacking.

Accounts	Prior Models	Primitive Models	Self-Repetition	Model Stacking
Active	2,055	3,416	24,465	13,512
Suspended	436	676	6,656	3,029

Rationale for the Primitive Models We selected this set according to three practical design principles derived from prior abuse reports [368] [108] and OSN policy guidelines[363].

(i) Single-edit fidelity. Each primitive makes at most one intuitive change—an insertion, deletion, or substitution—so the resulting string stays almost identical both visually and phonetically to the seed (e.g., ‘@BarackObama’→‘@BaarakObama’). This mirrors real-

world impersonation tactics, where attackers favour minimal edits that pass an unsuspecting user’s glance yet still satisfy username syntax rules. (ii) Broad character-class coverage. The palette spans vowels, doubled characters, digits, underscores, and common homoglyph confusions; introducing digits and underscores (four of the ten models) fills a clear gap left by URLCrazy [336] and AppCrazy [139], whose URL/package-name assumptions forbid such characters. (iii) Efficiency and noise control. Restricting ourselves to one application of a primitive keeps the candidate set tractable while maximising the chance that each variant will still ‘look right’ to a human; preliminary measurements (see Table 4.1) confirm that deeper edits yield sharply diminishing discovery returns and rapidly drift into obviously fraudulent territory. In sum, these ten primitives give broad yet disciplined coverage of plausible squats; because they are orthogonal, they compose cleanly under the self-repetition and stacking mechanisms, enabling UsernameCrazy to explore a far wider variant space without inflating the primitive set—and, thanks to their modular implementation, additional models can be incorporated (or obsolete ones removed) with minimal engineering effort.

4.2.4 Code Availability

The implementation of UsernameCrazy described in this chapter is available at a GitHub repository [173] under the MIT open-source license available at GitHub [174]. For instructions on how to run the code and generate squatted usernames based on a specific seed refer to ‘README.md’ file.

4.3 Measurement Methods

In this Section, we systematically explore the core research questions (MQ1–MQ4) of our research by exploring the prevalence, impact, characteristics and behaviour of username-squatted profiles on OSNs.

4.3.1 MQ1. Measuring Prevalence of Username Squatting

To answer our first MQ, we apply UsernameCrazy on the *Initial Seed* with the most popular X accounts [243] ($n = 97$) to generate their username variants. We focus on popular accounts, since these are more likely to be targeted by username squatting. Three independent raters went through the accounts and labelled them according to X 's accepted account types [360]: a) '*Government*' (5 accounts), b) '*Companies, brands and organizations*' (5 accounts), c) '*News organizations and journalists*' (10 accounts), d) '*Entertainment*' (59 accounts), e) '*Sports and gaming*' (15 accounts) and f) '*Activists, organizers and other influential individuals*' (3 accounts). Conflicts were resolved with discussion. The exact categorization can be found in Table 4.2. Then we search for each of the variants on the social network to find how many of them currently exist. We call such accounts *active*. To better understand if such username squatting can be exploited for malicious purposes, we use Twitter's Academic API v2 (**User Lookup** [362]) to check if any of the generated variants were already suspended.

Table 4.2: User categorization as determined by three independent raters following the official X documentation. The accounts '@realdonaldtrump', '@arianagrande' and '@aamir_khan' shown in light grey are not included in our experiments.

User Categorization	
Government: (5 accounts)	@barackobama, @narendramodi, @pmoindia, @potus, @hillaryclinton, @realdonaldtrump
Companies, brands, and organizations: (5 accounts)	@youtube, @twitter, @instagram, @nasa, @google
News organizations and journalists: (10 accounts)	@cnbrk, @nytimes, @CNN, @bbcbreaking, @sportscenter, @espn, @bbeworld, @theeconomist, @reuters, @natgeo
Entertainment: (59 accounts)	@katyperry, @justinbieber, @rihanna, @taylorswift13, @ladygaga, @theellenshow, @blakeshelton, @jtimberlake, @kimkardashian, @selenagomez, @britneyspears, @shakira, @jimmyfallon, @drake, @mileycyrus, @jlo, @brunomars, @oprah, @iamsrk, @srbachchan, @niallofficial, @ricky_martin, @beingsalmankhan, @kevinhart4real, @wizkhalifa, @liltunechi, @harry_styles, @louis_tomlinson, @akshaykumar, @liampayne, @pink, @chrisbrown, @onedirection, @aliciakeys, @kyliejenner, @kanyewest, @emmawatson, @conanobrien, @kendalljenner, @zaynmalik, @khloekardashian, @adele, @ihrithik, @actuallynph, @deepikapadukone, @pitbull, @danieltosh, @priyankachopra, @kourtneykardash, @shawnmendes, @coldplay, @bts_twt, @eminem, @arrahman, @nickiminaj, @avrilavigne, @mariahcarey, @davidguetta, @anushkasharma, @arianagrande, @aamir_khan
Sports and gaming: (15 accounts)	@cristiano, @neymarjr, @kingjames, @realmadrid, @imvkohli, @fcbarcelona, @sachin_rt, @kaka, @manutd, @nba, @championsleague, @nfl, @mesutozil1088, @andresiniesta8, @premierleague
Activists, organizers, and other influential individuals: (3 accounts)	@billgates, @elonmusk, @mohamadalarafe

4.3.2 MQ2. Measuring Squatted Account Effects:

To understand the impact of squatted accounts, we focus on features of OSNs that allow users to interact with and find other users. These are likely to be affected by username squatting issues since they depend on users typing usernames of other users, thus amplifying the impact of squatted accounts. In particular, we focus on the ability of users to *mention* and to *search* for other users. Both are popular features present in most OSNs.

User Mentions. We conduct experiments to measure whether username squatting affects the *mention* functionality. More interesting to our analysis are *typo-mentions*. We define *typo-mentions*, as a mention of a username which was made by mistake. Such typo-mentions can drive more activity to the squatted account, amplifying their potentially malicious impact. To measure such typo-mentions we use the full-archive search endpoint (API v2 [361]) to search for the most recent 500 tweets that *mentioned* each username variant. These tweets can be either actual tweets, retweets or replies and for each tweet, we collect the number of likes, retweets, quoted retweets and whether it can contain sensitive information such as links. Note that a *mention* of a username variant on its own does not necessarily indicate a typo-mention.

To further distinguish typo-mentions from purposeful mentions, we use the following observation: a user is more likely to have made a typo-mention, if the mentioner–mentionees are not directly related. Or conversely, if they are related, then a mention of a username variant was most likely on purpose. We can determine this relationship by looking at the distance between mentioner–mentionees in the social graph. In our experiments and due to API rate limits, we set this distance to ‘1’ hop.

User Search Recommendations. Most OSNs make recommendations to users trying to either *mention*, *tag* or *search* for another user. We design experiments to gain evidence on whether such recommendations can exacerbate confusion due to username squatting. In particular, we use the `GET users/search` [364] method of the X API, which takes as

input a prefix of a username and returns the first 1000 matching results. We apply this request for each of the possible prefixes of a username variant (if a username variant has n characters, we issue n requests). We repeat this experiment for each account of the *Initial Seed* to examine whether the generated username variants are ranked higher in the list of the recommended users than the respective original accounts and how often this phenomenon happens.

4.3.3 MQ3. Measuring Variant Characteristics

For MQ3, we aim to analyse the characteristics of squatted usernames to help us understand which ones are more prevalent or contribute to confusion the most. String generation models used in squatting methodologies can produce string variants that are significantly different from the seeded input but also among themselves. These can change the phonological properties of usernames, but we hypothesize that smaller changes can be more confusing as they can result in similarly sounding and similarly looking usernames. To measure this, we leverage a string similarity measure, and in particular on the edit distance (ED) between a squatted account and its seeded one.

4.3.4 MQ4. Measuring Account Behaviour

To understand whether squatted accounts exhibit suspicious behaviour we aim to identify i) how many accounts out of the generated ones have already been suspended, ii) how many accounts are bot accounts, iii) the behaviour of squatted accounts that share similar profile picture and name.

Suspended accounts. We begin with a measurement of how many of the squatted accounts, out of all the generated variants, have already been suspended by X . In detail, we use the Academic API v2 and we apply the `User Lookup` [362] method on every generated username. This call returns useful static information of an individual or a

group of accounts, such as the *name*, *username*, and the *creation date*. If an account is suspended, we conclude that these accounts were involved in some form of malicious (e.g., impersonation) or unwanted (e.g. confusion) behaviour. As these accounts no longer exist on the network, the *X* API does not allow us to gather the content of their profiles. To overcome this we make use of the *Wayback Machine* [152]³. Two independent raters then analyze the content of these profiles. We based the content analysis on *X*'s prohibited activity [324] and classified the profiles as i) impersonation attempts; ii) bots engaged in automated or spamming activity; iii) accounts who explicitly mention they are fan accounts, a parody or fake version of the celebrity account; this kind of 'impersonation' is allowed by *X*'s policy; iv) humans, i.e. users who are not related to the celebrity account except for sharing a username variant, v) other accounts which do not fall in the above categories.

Active account analysis. In order to better understand the behaviour of the active accounts on the network we first run Botometer [281] for *every* generated variant to understand how many of the accounts are bots. Botometer supports two types of scores, '*English*' and '*Universal*' (language independent). The former takes into account 6 categories of features, while the latter omits 2 of them (content and sentiment features) which are English-specific. Note that the '*Universal*' score is not as accurate as the '*English*' [260, 117]. Therefore, during our experiments we try to work with English-related users as much as possible. We note that we examine only a subset of our users because Botometer does not return a score for locked accounts or for profiles with no activity.

To gain a deeper understanding into the activity of the active profiles, we perform a manual analysis of the activity of a number of randomly selected squatted username accounts that share a similar username with a celebrity and have a profile picture. Two independent raters then analyse the behaviour of the accounts. If an account had no suspicious behaviour it was labelled as *benign*. If an account can contribute to confusion

³The snapshots for which we get a '*Got an HTTP 302 response at crawl time*' error are excluded from the analysis.

we term it *suspicious*; these were subdivided into the following: a) *impersonation*: an account that has no specific goal other than impersonate, e.g. this impersonation could be done for entertainment purposes or the creator of the account could be a fan, but does not explicitly mention so on the account; b) *financial* accounts that use the generated confusion for a financial incentive, e.g. they try to sell products or gain followers; c) *political* aiming to use the confusion to influence their followers politically, e.g. such accounts may (re)post news or opinion articles; d) *news* accounts that spread (fake) news about the celebrity they impersonate, these accounts differ from the impersonation accounts as they have the specific goal of spreading news about the seed account holder; e) *harass*: accounts that directly harass (e.g., insult or aggressively mention or retweet) the celebrity they impersonate or other celebrities; and f) *other*: other accounts that did not fall into the other categories, e.g. accounts with little or unrelated activity.

Amplification of confusion. Crucial to our analysis is to understand whether squatted accounts with malicious purposes employ other similarities with the original account. To measure this we search for active squatted accounts with similar *profile names* and *profile images* to the original and analyse how many of those accounts are bots. We apply VGGFace2 [45], a popular image recognition model (see Section 5.2 and Algorithm 1 for details on the algorithm) on all squatted usernames⁴ and then we compare, using the Levenshtein distance [183], their profiles names with their potential target. We consider profile names to be similar if either i) two names are equivalent, ii) an extra character is added as a prefix or suffix, iii) a substring of the original name appears as a substring in the squatted name (e.g., for ‘Cristiano Ronaldo’ we will check separately whether ‘Cristiano’ or ‘Ronaldo’ appear in the squatted account’s profile name). Details of the algorithm can be found in Algorithm 2. Out of those accounts which share similar image and/or name, we compute how many are bots using Botometer.

⁴We examine the squatted accounts which have been generated from the original users of the ‘Entertainment’, ‘Government’, ‘Sports and gaming’ and ‘Activists, organizers, and other influential individuals’ category. Also, profile images that do not depict a face are excluded.

Algorithm 1 Image Similarity Pseudocode

Input All usernames (original + generated)**Output** Similar images

```

1: procedure IMAGE_SIMILARITY
2:   users[]  $\leftarrow$  original usernames / initial seed
3:   if user not a celebrity then
4:     Remove user from users
5:   for user in range(len(users)) do
6:     orig_img[user]  $\leftarrow$  download image of celebrity
7:   for user in range(len(orig_img)) do
8:     if image[user] does not have a face then
9:       Remove orig_img[user] from orig_img[]
10:      Remove user from users
11:   gen_users[][]  $\leftarrow$  generated username variants (UVs)
12:                                      $\triangleright$  1st array: all the remaining users
13:                                      $\triangleright$  2nd: all the generated UVs of a user
14:   for i in range(len(gen_users)) do
15:     for j in range(len(gen_users[i])) do
16:       gen_imgs[i][j]  $\leftarrow$  download image of user
17:
18:                                      $\triangleright$  there are multiple images in each row
19:   for i in range(len(gen_imgs)) do
20:     for j in range(len(gen_imgs[i])) do
21:                                      $\triangleright$  calculation of distance between embeddings
22:       if distance(orig_img[i], gen_imgs[i][j]) < threshold then
23:         return Pair of similar images
24:       else
25:         No similarity

```

4.4 Characterization

In this section we present our measurement results from i) measuring the prevalence of username squatting, ii) identifying through what ways squatted accounts can contribute to online confusion, iii) determining the characteristics of the squatted usernames driving this online confusion, and iv) discovering the behaviour of the suspended and bot accounts.

Algorithm 2 Profile Name Similarity Pseudocode

Input All usernames (original + generated)**Output** Similar profile names

```

1: procedure PROFILE_NAME_SIMILARITY
2:   users[]  $\leftarrow$  original usernames / initial seed
3:   gen_users[][]  $\leftarrow$  generated UVs ▷ 1st array: all the users
4:   ▷ 2nd: all the generated UVs of a user
5:   for i in range(len(gen_users)) do
6:     try
7:       ▷ For space limitations we assume that a profile name can be splitted by “ ”
8:       prof_name  $\leftarrow$  gen_users[i][0].split(' ')
9:       splitted_name  $\leftarrow$  gen_users[i][0].split(' ')
10:      for j in range(1, len(gen_users[i])) do
11:        if prof_name == gen_users[i][j] then
12:          return Exact match
13:        else if gen_users[i][0] in gen_users[i][j] AND
14:          gen_users[i][j] not in prof_name then
15:            return Exact match plus at least one
16:              character extra
17:        ▷ For space limitations we assume that a profile name consists of 2 words
18:        else if splitted_name[0] in gen_users[i][j] AND
19:          splitted_name[1] not in gen_users[i][j] then
20:            return At least one word of the prof_name
21:              is appeared but in a different order
22:        else
23:          return Other
24:      catch ▷ We treat the profile name as one word only
25:        for j in range(1, len(gen_users[i])) do
26:          if prof_name == gen_users[i][j] then
27:            return Exact match
28:          else if gen_users[i][0] in gen_users[i][j] AND
29:            gen_users[i][j] not in prof_name then
30:              return Exact match plus at least
31:                one character extra
32:      endtry

```

4.4.1 Prevalence of Username Squatting.

We conducted our **MQ1** measurements on X , in October 2021 and found a total of 41,393 active username variants from the *Initial Seed* ($n = 97$) of popular accounts, or approximately 427 variants per username. Table 4.3 lists the five top original usernames with

the most active username variants. We also found that 10,361 accounts with username variants were suspended, or 107 on average per original account. This is a good signal that in the past these accounts were malicious [357] which we further analyse in Section 4.4.4. Next, we examined how these suspended accounts are spread across different categories. We found that 7,940 accounts (66.3%) belong to the ‘Entertainment’ category, 1,175 accounts (9.8%) in ‘Sport and Gaming’, 1,104 accounts (9.2%) in ‘Companies, Brands and Organizations’, 709 accounts (5.9%) in ‘Government’, 582 accounts (4.8%) in ‘Activists, Organizers and Other Influential Individuals’ and 460 accounts (3.8%) in ‘News Organizations and Journalists’ category. In May 2022, we ran again the measurements and found 43,349 active and 11,384 suspended accounts while in November 2022 we found 43,994 active and 11,827 suspended accounts (see Table 4.3). Alarming, the increase on the number of the suspended users in every new snapshot indicates that this is an ongoing and growing problem. For the rest of the Chapter, we use the first snapshot. Therefore, we conclude that celebrities and popular users in general are indeed a common target of username squatting attack (see Section 5.5), something that contrasts the insights of previous work [108] which found 166 potential impersonation pairs and only 3 of them were celebrity impersonation attempts within a three month period (as previously stated in Section 3.1).

This discrepancy stems from four restrictive assumptions in [108]: (i) *Manual candidate selection and verification*. All look-alike profiles had to be *manually* shortlisted, and for ambiguous cases the authors even contacted account owners to ask whether they were aware of the similar profile—an approach that is labour-intensive, slow, and inherently low-recall when applied to a platform the size of X; (ii) *Local-neighbourhood graph traversal*. Their detection stage depended on costly breadth-first searches restricted to the immediate social-graph neighbourhood of a seed account, implicitly assuming that an impersonator (or cluster of bots) is at most a few hops from its victim—an assumption broken by isolated or dormant squatted handles; (iii) *Interaction interpreted as benign*. Any pair of accounts exhibiting *any* interaction (follow, mention, like, reply) was labelled

a harmless “avatar–avatar” case and excluded. In contrast, we observe that 521/540 confirmed malicious impersonators in our dataset fall precisely into this category, showing that interaction is not a reliable safety signal; (iv) *Reliance on graph proximity over lexical cues*. The Doppelgänger pipeline pivoted on social-graph proximity rather than systematic exploration of *username variants*, overlooking attackers that operate in isolation or that squat a handle long before accumulating neighbours.

By eliminating these constraints—removing manual vetting, eliminating graph-distance constraints, and continuing to examine interacting pairs—and adopting a username-centric, graph-agnostic methodology—our framework uncovers a far larger and more diverse set of impersonation attacks, thereby demonstrating that high-profile users are systematically targeted on X.

Table 4.3: Top five original accounts with the most active username variants on X. ‘Generated Accounts’, ‘Active Accounts’ and ‘Suspended Accounts’ are abbreviated as ‘*Gen. Acc.*’, ‘*Act. Acc.*’ and ‘*Sus. Acc.*’, respectively. Also, we denote the ‘Generated/Active/Suspended Accounts with an edit distance 1-3’ as ‘*Gen./Act./Susp. Acc. ED 1-3*’.

Potential Squatted Usernames						
Original Accounts	Gen. Acc.	Gen. Acc. ED 1-3	Act. Acc.	Act. Acc. ED 1-3	Sus. Acc.	Sus. Acc. ED 1-3
@kaka	5,899	1,380	2,145	441	224	96
@cristiano	17,687	2,301	1,714	639	148	63
@pink	11,955	1,348	1,555	381	162	75
@nasa	5,842	1,356	1,534	333	231	42
@selenagomez	9,925	2,270	1,346	306	177	97
Total (Oct. 2021)			41,546	37,965	10,208	9,762
Total (May 2022)	851,682	378,088	43,349	39,403	11,384	10,781
Total (Nov. 2022)			43,994	40,200	11,827	11,133

4.4.2 Online Confusion.

User Mentions. For **MQ2**, by applying the methodology described in Section 4.3.2 we collected 1.2 million tweets to measure the contribution of squatted accounts to online confusion. Tweets can either be *actual tweets*, *retweets*, or *replies* that mention at

least one of the generated username variants. We found that out of the 864,369 actual tweets, 657,337/864,369 (76%) are likely *typo-mentions* and only 200,562/864,369 (23.2%) happen when the mentioner and mentionee are within one hop on the social graph (*purposeful-mentions*). We discard the rest of the tweets (6,470 or 0.8%) because the API did not return information about the friendship of the users.

We further analyze the 657,337 *typo-mentions* and found that 437,550 tweets (66.6%) belong to the ‘Entertainment’ category, 96,720 tweets (14.7%) in ‘Sport and Gaming’, 57,126 tweets (8.7%) in ‘Companies, Brands and Organizations’, 33,220 tweets (5.1%) in ‘News Organizations and Journalists’ category, 25,206 tweets (3.8%) in ‘Government’ and 7,515 tweets (1.1%) in ‘Activists, Organizers and Other Influential Individuals’. We observe that typo-mentions are prevalent in all categories but most are found in Entertainment accounts as expected since our dataset contains more such accounts. Typo-mentions are frequent which is worrisome since they can drive more traffic to the suspicious account.

User Search Recommendations. In order to identify the factors that can lead to the increased popularity of potential squatted accounts, we take a closer look at the suggestion algorithm of X. We expect that when a user searches for a celebrity using the search bar the targeted profile should be returned within the first 10 options due to its popularity. Using the methodology from Section 4.3.2, out of the 929 requests (the total number of the characters of the usernames of our *Initial Seed*) we find that only 166 and 178 times all the original users together appear in the first 10 and 100 recommendations, respectively. Alarming, only when the full original usernames were inserted in the search bar, the official profiles were returned as the first 15 options (see Table 4.4). We observed that there exist instances where the squatted username appears *before* the original account. Clearly, this is problematic as it can exacerbate the problem of online confusion. More precisely, 1,205 of active squatted accounts appear in the first 1,000 recommendations, 705 of them appear in the top 500 recommendations, 170 in the top 100 and 26 appear in the top 10 recommendations while searching for our seed accounts (see Figure 4.4). This indicates that the popularity of squatted usernames can be amplified through the search

algorithm of the platform itself. Moreover, we manually check the 170 squatted users appearing in the top 100 recommendations and marked 10% as impersonators and 38.8% as malicious. This shows alarmingly that search recommendations can not only amplify confusion but can also be leveraged by malicious actors.

Table 4.4: Original accounts within the first 10 and 100 search recommendation results from a total of 929 requests.

Original Accounts - X 's Recommendation System		
Position	# of Times Original Accounts Recommended	Observations
Top 10	166	108 times the original accounts returned as first recommendation but only when their full name was inserted in the search bar ($ED = 0$)
Top 100	178	132 times the original accounts returned in position 11 to 15 but only when their full name was inserted in the search bar ($ED = 0$)

4.4.3 Squatted Username Characteristics.

For **MQ3**, we aim to analyze the characteristics of squatted usernames to help us understand which ones are more prevalent or contribute to confusion the most. To do that we use the edit distance measure (see Section 4.2). The highest number of suspended and active accounts are squatted usernames where the edit distance is between one and three characters (see Figure 4.5). Looking into the high numbers for $ED = 3$, we found that in numerous cases, username variants consist of an addition of *three* digits before or after the seed username.

Making 1-3 mistakes when typing a word is reasonable [69, 107]. Figure 4.6 further depicts that i) indeed users often make 1 to 3 mistakes in their mentions and ii) most of the times there is no relationship between the mentioner and the mentionee. The number of mistakes in actual tweets is more informative than in replies or retweets, as the latter can simply propagate the same typo mistakes. In addition, Figure 4.4 illustrates the effects of the edit distance on X 's search recommendations. Out of the squatted accounts suggested by

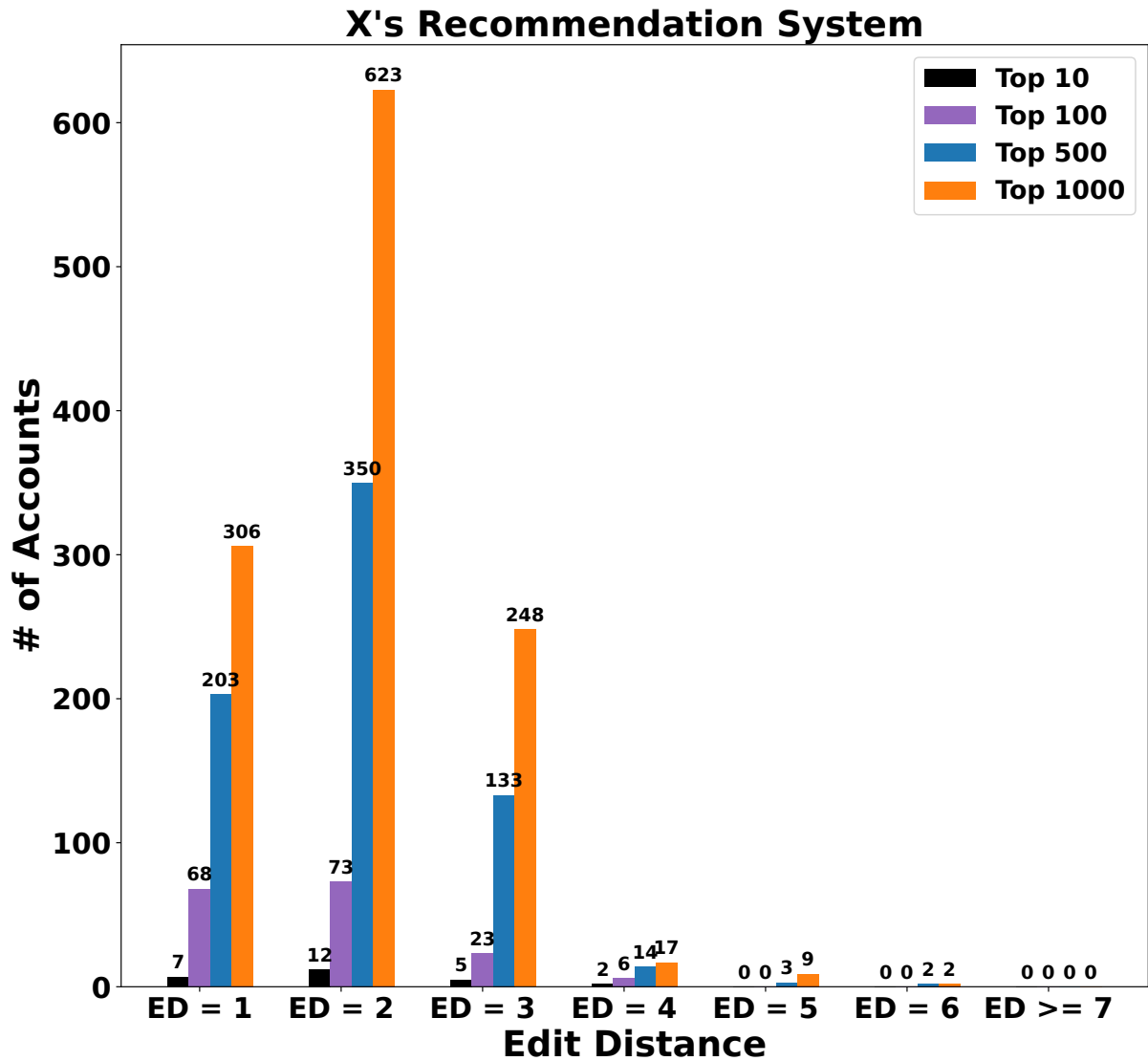


Figure 4.4: Squatted accounts in the top search recommendations grouped by edit distance.

the search algorithm, again we note that those with edit distance 1-3 are most common. Interestingly, 24 squatted usernames appear within the top 10, 164 within the top 100 and 686 within the top 500 of the search recommendations. From these results, we can conclude that an adversary can identify specific patterns of typo mistakes and claim a username that is close to the original. Our results indicate that typo-mentions and the search algorithm can be exploited to increase traffic to squatted accounts.

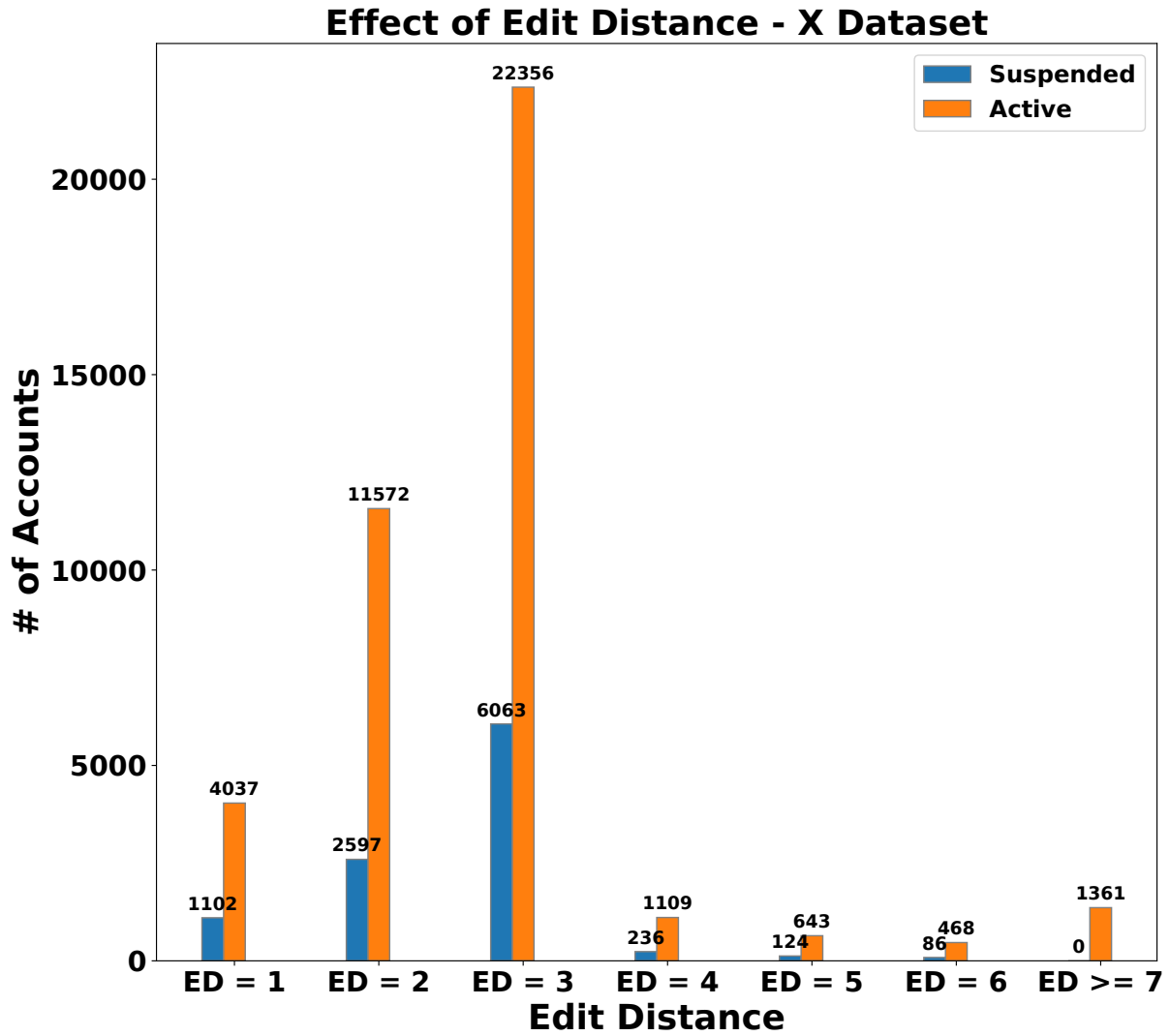


Figure 4.5: The number of active and suspended accounts across different edit distance categories.

4.4.4 Squatted Account Behaviour.

For **MQ4**, we aim to analyse the activity of both suspended and active accounts. For this we manually analysed the behaviour of 135 suspended accounts using the Wayback Machine, we classified the behaviour of 540 suspicious and active accounts and we applied Botometer to all active accounts.

Suspended Accounts Analysis. By applying the methodology described on Section 4.3.4 we found 10,208 suspended accounts. *Wayback Machine* [152] returned valid

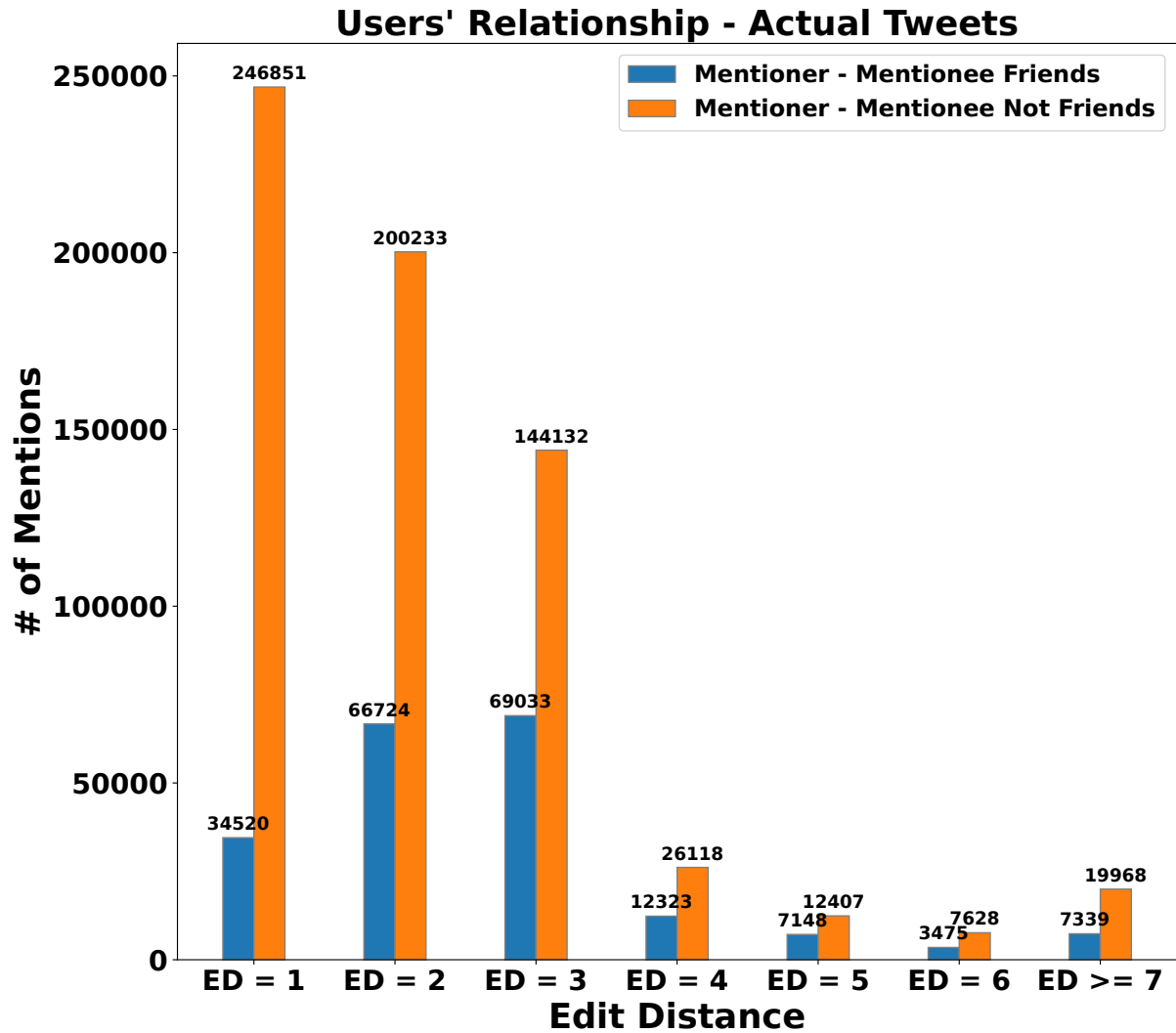


Figure 4.6: Number of actual tweets that mention at least one username variant grouped by the variants' edit distance with their original accounts.

snapshots for 135 of those accounts⁵. Then, two independent raters went through all the available snapshots and they found (see Table 4.5): 52 impersonation attempts of the original account, 12 accounts posting e.g., spam links or product promotions, 6 human accounts that explicitly mention they are fan or parody accounts, 15 profiles of humans with no connection to the original account, 45 other accounts. Other include private profiles or accounts with no activity. Also, for 5 accounts there was a disagreement between the raters so we do not report their results. This analysis provides a first validation of our idea that a suspension was the consequence of malicious content creation.

⁵For the rest 10,226 accounts, Wayback Machine returned no or corrupted snapshots.

Table 4.5: Manual classification of suspended accounts.

Suspended Accounts (135 users)				
Impersonations	Bots	Fans	Humans	Other
52 (38.5%)	12 (8.8%)	6 (4.4%)	15 (11.1%)	45 (33.3%)

Active Accounts Analysis. Previous works [32, 73] used Botometer as ground truth for measuring the bot activity of accounts. To validate our hypothesis that many of the squatted usernames are indeed malicious and/or bots, we applied Botometer to all the generated usernames that are still *active* on *X*. Botometer uses *CAP* score, a probability which indicates that a profile with this score or greater is controlled by a software (i.e., is a bot). We set *CAP* score to 0.95, which is a conservative value expected to yield only 5% false positives [281]. Our analysis shows that on our dataset this value can be around 11%. Out of 41,546 variants Botometer returned a score for 25,089 since the other accounts are private profiles. 9,702 (38.7%) profiles were classified as automated users. The automated accounts are further categorized as different kinds of bots and the largest category was *fake followers*. We observe that a large number of the squatted users are malicious bots, indicating also the effectiveness of UsernameCrazy.

We next performed a manual analysis (see Section 4.3.1) to better understand the *type* of activity the squatted accounts are involved in. Out of the 41,546 active profiles we randomly selected 1,400 users who have a face or an avatar in their profile picture. Two raters agreed beforehand on the rules they will follow, discussing about the characteristics of impersonators, spam/bot and benign users. Accounts that had no similarities with the seed account or explicitly say they were fan or parody profiles were labelled as *benign* [365], while accounts that were sharing same features with their original accounts were labelled as *suspicious*. Also, profiles sharing a number of features⁶ as their seed account but with limited or no activity were labelled as *suspicious*. The process allowed us to identify 838 accounts as *benign*, 540 as *suspicious* (defined as an account that can contribute to confusion) username squatting attempts and 22 accounts were left as ‘difficult to declare’.

⁶Profile name or picture, username.

We examined the inter-rater agreement using Cohen’s Kappa [217] and found $\kappa = 0.92$ and $\kappa = 0.89$ for the benign and suspicious users, respectively, which indicates *near perfect* agreement. Then, the two manual raters further categorized the suspicious accounts’ behaviour as *impersonation*, *financial*, *political*, *news* and *harass*. The full results for the suspicious accounts are presented in Table 4.6. We conclude that a large amount of active accounts are involved in some form of malicious or confusing behaviour.

Table 4.6: Manual classification of the active malicious accounts used in dataset.

Active Malicious Accounts (540 users)					
Impersonations	Financial	Political	News	Harass	Other⁷
115 (21.3%)	56 (10.3%)	7 (1.3%)	15 (2.6%)	4 (0.7%)	343 (63.5%)

Analysing Confusion Amplifiers. Using the method presented in Section 4.3.4 we analyse how many bot accounts share similar features (here profile name/image) with the original one. Figure 4.7 depicts that indeed a large number of users have a similar name/image with their target; this suggests that these profiles likely impersonate the original owner to drive traffic to their account and increase the impact of their malicious activity.

4.5 Summary

This Chapter investigated the phenomenon of username squatting on X, examining its prevalence, impact and associated behaviours. To facilitate our study, we developed UsernameCrazy to generate username variants. The findings revealed that username squatting is popular towards popular accounts. We also found that squatted variants contribute to online confusion, a problem further amplified by platform features like mentions and search recommendation. The analysis categorised squatted accounts into benign, confusion and impersonation attempts. Furthermore, many suspended accounts were found to have malicious intent. In addition, the study identified common squatted patterns used across

⁷Accounts with little or unrelated activity.

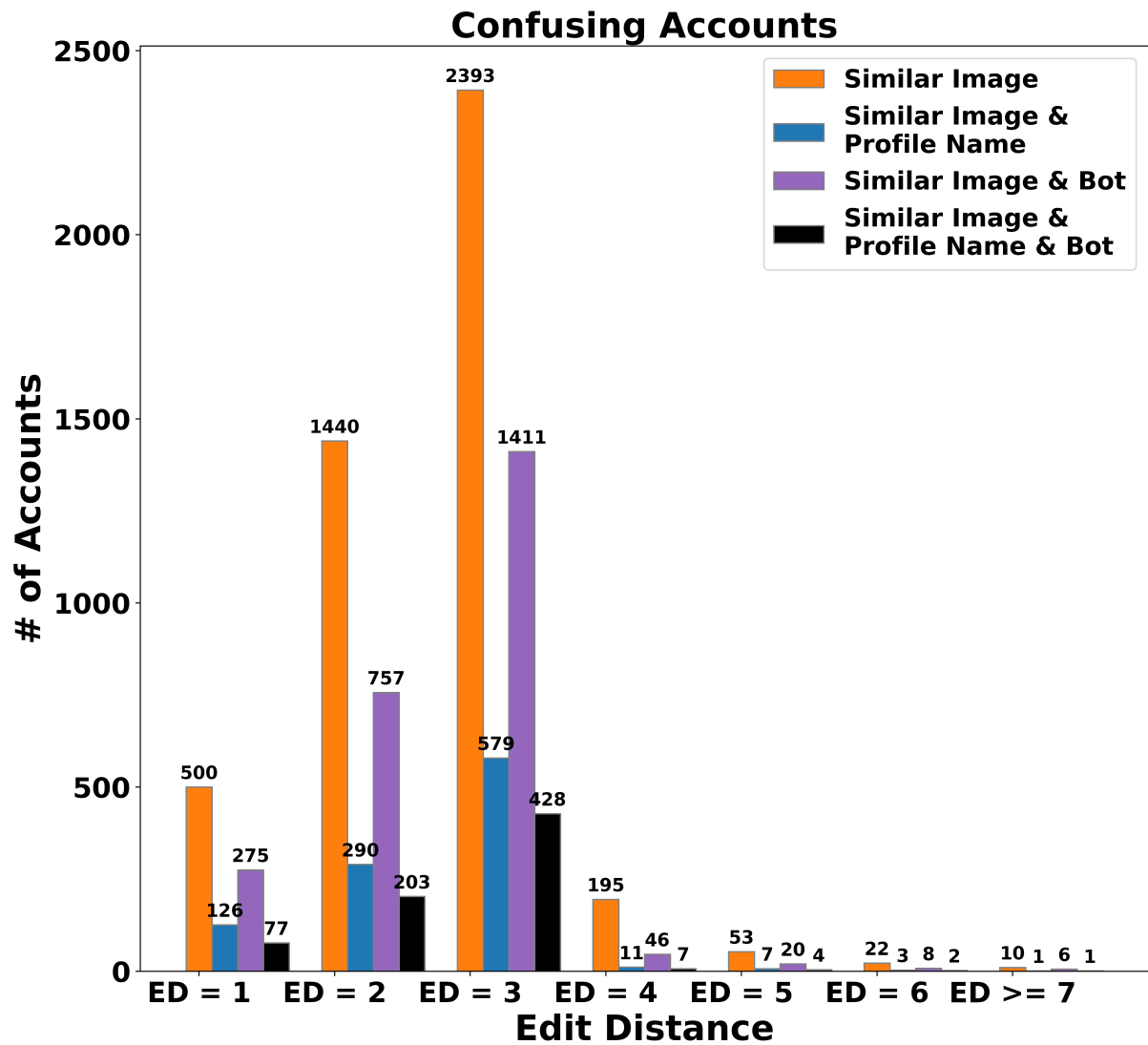


Figure 4.7: Classification of the confusing accounts.

the network, like small edits of 1 to 3 characters between usernames, like the number addition. Lastly, a combination of manual review and automated bot detection showed that there is a significant malicious activity among active squatted accounts, highlighting the need for better detection frameworks to address this phenomenon.

Chapter 5

An Impersonation Detection Framework for X: Design and Evaluation

This Chapter presents the design and methodology of SQUAD, a framework for detecting suspicious squatted variants on *X*. SQUAD aims to efficiently distinguish benign from potentially harmful accounts. The framework generates squatted usernames using UsernameCrazy, filters active accounts, and then extracts features for classification. SQUAD effectively minimizes false positives while flagging accounts for further scrutiny, enhancing the detection of impersonation and confusion behaviors.

5.1 SQUAD Design Overview

Our measurement study revealed that username squatting is an important issue on *X* as a significant subset of these accounts are bot accounts or impersonations involved in

malicious behavior. On the other hand, a large number of squatted accounts are benign (e.g., fan and parody accounts) and we need an effective way to make that distinction. This raised the following question: *Can we design an effective and efficient framework for detecting suspicious username squatting attempts?*

Toward this end, we developed a novel methodology embodied in an end-to-end username **SQU**Atting **D**etection framework (SQUAD) to facilitate the detection of potentially malicious confusion attempts against popular accounts. To do that, SQUAD uses UsernameCrazy in combination with a classifier to output accounts that are confusing and require further analysis.

Figure 5.1 illustrates the overall architecture of the framework. First, the seed account for which one wants to find confusing versions is selected. This is used as input in UsernameCrazy for the generation of squatted usernames (see Section 4.2.3). These are fed into a *filtering* component which identifies active accounts corresponding to the squatted usernames¹. The active accounts and the *Initial Seed* account, are fed into a *feature extraction* component which extracts certain account features. These are then passed into the *classification* component. This outputs pairs of $\langle \text{initial}, \text{squatted} \rangle$ accounts where the squatted account is sufficiently confusing and requires further analysis. The username generation and filtering components were introduced during our measurement study. Next, we focus on the feature extraction and classification.

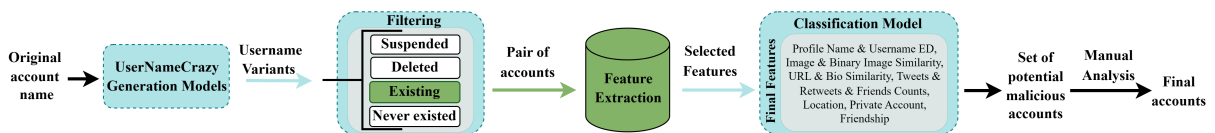


Figure 5.1: High-level architecture of SQUAD.

¹The filtering also identifies suspended, deleted and non-existing accounts which might be useful for further analyses.

5.2 Feature Extraction and Selection

Initial Feature Set. We start with an overview of the account features our classifier used. We manually select the features that could play a role in the account being confusing. This selection was based on our observations and findings from prior works identifying impersonation attempts and bot accounts [86, 237, 281, 108].

- *Image Similarity Score.* We use an image recognition model, VGGFace2 [45], to identify whether the squatted account’s profile image is of the same person as the seed account’s. It makes use of a convolutional neural network, known as SE-ResNet-50 [138, 45]. To ensure accuracy of the model on our dataset we first performed a test on a manually labelled subset of 128 images²; this resulted in a 85.93% test accuracy for the cutoff threshold 0.65. Table 5.1 depicts the accuracy results of VGGFace2 with different thresholds. Note that all images with a higher score are classified as ‘*no similarity*’). Also the test accuracy is lower than previously reported (90.8%) [45] but still high enough to justify using this model for our prototype implementation.

Table 5.1: Accuracy of VGGFace2 with various thresholds. Setting the threshold value to 0.65, the model has the best accuracy.

VGGFace2				
Threshold	Accuracy	# of Images	# of FP	# of FN
0.5	82.81%	128	9	13
0.6	84.37%	128	8	12
0.65	85.93%	128	6	12
0.7	85.15%	128	7	12

We present the detailed methodology that was used for selecting and comparing the images automatically for a large number of accounts in Algorithm 1. The probability output from this model is then used as the image similarity score.

²We removed accounts that did not represent a human (e.g., companies and teams) or that portray groups of people, like bands. As a result, we are left with 64 accounts. For each of these accounts we pick 2 random generated variants and their profile pictures, resulting in 128 profile photos in total.

- *Binary Image Similarity*. This feature is a binary variant of the aforementioned one. If the similarity score is above the selected threshold, its value is ‘no’, else ‘yes’.
- *Profile Name Edit Distance*. We compute the Levenshtein distance [183] between two profile names. We then use the raw output of the edit distance.
- *Username Edit Distance*. We compute the Levenshtein distance [183] between two usernames. We then use the raw output of the edit distance.
- *Friends count*. The number of friends an account has.
- *Followers Count*. The number of followers an account has.
- *URL Similarity*. To calculate URL similarity we collect the textual representation of a profile’s ‘website’ field for both accounts. We first examine whether the URLs are exactly the same or the squatted account uses a URL that includes the seed username as a substring of it. If any of these is true, the result is ‘1’. If the accounts use completely different URLs the result is ‘0’.
- *Bio Similarity*. For bio similarity we use the *Jaccard Distance* [77] which is a similarity metric between two sets. The similarity index ranges from 0 to 1. The closer to 1, the more similar the bios. Even though *Jaccard similarity* does not have a notion of semantics, we found it to perform comparably with calculating the distance between the embeddings of bio statements, where we calculated 200-dimensional embeddings pre-trained on Tweets (Global Vectors for Word Representation embeddings [250]). We also convert any special character (e.g., emojis) in to word format.

We opted for token-set Jaccard similarity because typical Twitter bios are short (median length ≤ 18 tokens)³ [288] and often differ only by a handful of surface terms (@-handles, hashtags, emojis) [245] [187]. In such cases an n-gram-overlap measure gives high interpretability (‘three of four words match’) and can be computed in $O(|A| + |B|)$ time with

³With the usual 4~5 characters [234] + space per token, that corresponds to roughly 15–18 tokens.

no model calls. Additionally, prior work on profile matching and duplicate-user detection already relies on it—for instance, Shoeibi et al. [295] apply Jaccard to audience–network sets when clustering nearly 20K Twitter accounts, reporting 97% classification accuracy, and Yang et al. [370] adjust Jaccard for short-text distance in social-media retrieval tasks.

To situate this choice, we next contrast Jaccard with two widely used alternatives—ROUGE and LLM-based evaluator methods. More specifically, ROUGE [191] was designed for multi-sentence summaries and its correlation with human judgements drops markedly on short texts—Liu et al. [192] report notably low correlation for meeting snippets that are only a few sentences long. Moreover, larger-scale evaluation frameworks that treat a large language model as the judge have recently been shown to suffer from measurable biases (e.g. misinformation-oversight, gender), raising reproducibility concerns[50][167].

Given that bios are (i) small, (ii) often differ only in surface identifiers, and (iii) need to be compared across hundreds of thousands of users, Jaccard therefore provides a straightforward and interpretable compromise: it is proven effective on Twitter data, robust for short strings, and avoids the bias and length-sensitivity issues that can affect other methods.

- *Friendship*. This feature shows if a squatted account follows the seed account. The return result is binary.
- *Location*. For every account, we collect the textual representation of the mentioned location, like the city name. We then check whether a squatted account shares the same location with the seed one. If yes, the returned results is ‘1’ (this is applicable also if both locations are empty). Otherwise, the result is ‘0’.
- *Private*. Indicates whether an account is private.
- *Tweet Count*. The number of tweets a user has.
- *Retweets Count*. The number of retweets a user has.

- *Tweet Sentiment.* For analyzing tweet sentiment we fetch the 500 most recent tweets of the squatted account. We use NLTK [35] for preprocessing the text to remove the stop-words, user-mentions and punctuation marks, and then we apply TextBlob [198], a sentiment method for classifying the tweets as *neutral*, *positive* and *negative*. For the accounts with no tweets, the score of every category is ‘0’. Since in almost 95% of the cases the returned result was ‘neutral’, we discard this feature.

Note also that we normalize all the data between 0 and 1 based on the min-max normalization process [348].

Feature Selection. The initial set of features showed success in relevant classification tasks but not all of them are necessarily important for our task. In fact, redundant features can decrease the performance of the classifier by introducing noise [42, 266]. To remove redundant features and improve the performance of the SQUAD classifier we select the optimal feature set via the *Recursive Feature Elimination* (RFE) with Cross-Validation process [248, 52] by optimizing for ‘accuracy’. Our final feature set includes the *Profile Name Edit Distance*, *Username Edit Distance*, *Image Similarity Score*, *Binary Image Similarity*, *Friends Count*, *Friendship*, *Tweet Count*, *Bio Similarity*, *URL Similarity*, *Location*, *Retweets Count* and *Private Account* features.

5.3 Dataset Generation

For each <seed, squatted> account pair, the extracted features will be used in a binary classification whose output is the label *benign* or *suspicious*, where suspicious refers to an account that can contribute to confusion. The goal of a successful classifier is to have a low false positive (a positive being a suspicious account) rate, so that when the tool is used in combination with manual analysis it allows to find malicious activity on the OSN as efficiently as possible.

A main challenge in our framework’s classification modeling is that there are no pub-

licly available datasets of impersonation attempts which could use to train and evaluate candidate models. Existing ones, such as [333, 200, 23], are either datasets that consist of bots, or fake accounts which are not trying to imitate a specific profile. Our definition of confusion requires a different dataset as it encompasses any account that shares a sufficient number of features with the seed account including bots, impersonations and other accounts involved in malicious activity. Therefore, we created our own dataset by manually labelling a subset of the data records we collected for our measurement study. We use the same 1378 users which two raters manually and independently label as *benign* or *suspicious* (see Section 4.4.4). To fix the imbalance of our dataset, we produce more examples from the minority class by applying the Synthetic Minority Over-sampling Technique (SMOTE) [49]. We then use our labelled accounts to train and evaluate 7 popular binary classification models which our prototype implementation of SQUAD supports. In Section 5.4 we compare their performance.

5.4 Evaluation of SQUAD

The goal of SQUAD is to speed up the process of finding malicious activity on the network by identifying confusing accounts that require further manual analysis. It uses Username-Crazy to avoid searching the entire social graph by *reducing* the search domain. The filtering component helps identify from that username search domain which variants exist on the network. Here, we first determine *the effectiveness and efficiency of Username-Crazy in identifying squatted accounts and how it compares with previous string squatting tools* (Section 5.4.1). Then we analyze *the accuracy of SQUAD in classifying confusion accounts* (Section 5.4.2).

5.4.1 UsernameCrazy Performance

Effectiveness. First we evaluate the ability of UsernameCrazy to produce *relevant* username variants and compare it with prior tools which leverage squatting techniques for generating complex strings, namely *URLCrazy* [336] and *AppCrazy* [139]. We use the 10 most popular users of our ‘*Initial Seed*’ as input to the three generation tools and then apply our filtering component. We report the number of username variants produced, the number of those that actually exist on the network, the number of produced username variants that are already suspended by the network and the number of the existing impersonation attempts found based on our manual analysis. Table 5.2 summarizes our results with a full break down per user.

Table 5.2: Competency of *URLCrazy*, *AppCrazy* and *UsernameCrazy*.

Usernames	URLCrazy				AppCrazy				UsernameCrazy			
	# of Generated Usernames	# of Active Squatted Usernames	# of Suspended Squatted Usernames	# of Active Impersonators	# of Generated Usernames	# of Active Squatted Usernames	# of Suspended Squatted Usernames	# of Active Impersonators	# of Generated Usernames	# of Active Squatted Usernames	# of Suspended Squatted Usernames	# of Active Impersonators
@barackobama	135	20	6	0	57	35	12	0	9,800	307	104	1
@katyperry	118	17	9	0	25	15	5	0	5,992	640	500	7
@justinbieber	170	34	4	3	60	52	7	5	9,010	555	110	20
@rihanna	94	21	3	0	38	32	2	1	13,086	1,113	199	9
@taylorswift13	172	22	9	2	36	25	8	2	5,766	102	23	5
@cristiano	131	34	1	0	49	46	1	0	17,687	1,714	134	1
@ladygaga	96	18	2	0	32	26	6	0	6,821	954	287	10
@theellenshow	168	5	6	1	51	13	8	2	6,999	46	62	7
@youtube	101	30	2	3	41	31	9	5	13,057	1,078	353	12
@jtimberlake	156	15	8	1	47	18	15	2	8,750	120	129	16
Total (10 users)	1,274	215 (16.87%)	49 (3.84%)	10	436	293 (67.20%)	73 (16.74%)	17	96,968	6,629 (6.83%)	1,901 (1.96%)	88

We observe that UsernameCrazy exhibits better username generation coverage compared to existing methods. It generated $O(10^4)$ usernames compared to $O(10^3)$ and $O(10^2)$ from *URLCrazy* and *AppCrazy*. This is a result of UsernameCrazy’s enhancements with new models, *model self-repetition*, and *model stacking*. We also observe that compared to both prior methods UsernameCrazy identified one order of magnitude more variants that exist on the network and that are suspended. This showcases UsernameCrazy’s effectiveness in producing relevant string variations. There are several reasons for this. Firstly, OSNs have constraints on the number and type of characters a username can have. Prior tools do not take that into account and use generation methods (such as bit flipping or string rearrangement) which produce a large number of usernames that are invalid to begin with. Further, other tools produce strings that include a top-level domain either

at the head or tail of the name making them again automatically incompatible. Note that for UsernameCrazy the ratio of existing or suspended usernames over all generated usernames is low. This is because a lot of the usernames are valid but have not been used yet. Nonetheless, it is important to be able to produce them to discover future instances of squatting username attempts.

In contrast, it is important to recognize that existing tools like URLCrazy and AppCrazy follow different input and generation assumptions that make the comparison with UsernameCrazy non-trivial and influence their respective performance profiles. Specifically, URLCrazy mandates domain-like inputs (e.g., ‘cristiano.com’) which, by construction, produce output strings that include top-level domains (e.g., ‘.com’, ‘.eu’)—characters (like the ‘.’) that are not permissible in usernames on platforms like X (formerly Twitter) or Instagram. To ensure a fair evaluation in our study, we adapt this behaviour by first appending a generic domain suffix (e.g., ‘.com’) to seed usernames to satisfy URLCrazy’s requirements and then strip away domain components from the generated outputs to retain only valid username-like substrings. This two-step adjustment is essential to bring URLCrazy’s results into a comparable format.

AppCrazy, on the other hand, does not require a domain-formatted input and operates directly on application names; however, its string generation logic is constrained and optimized for app-store contexts. Notably, AppCrazy generates squatting variants by applying rules such as keyword overlap, brand extension (e.g., ‘FacebookLite’, ‘InstagramPro’), or affixation, based on naming conventions commonly observed in mobile app marketplaces. While this is suitable for detecting malicious apps on mobile platforms, these patterns are structurally distinct from the ones used by impersonators on OSNs who tend to favour minimal edits or number insertions to usernames. Furthermore, AppCrazy relies heavily on curated seed lists and rules derived from the app name ecosystem, limiting its adaptability across domains. Unlike UsernameCrazy, URLCrazy and AppCrazy do not incorporate a composable or extensible model architecture (e.g., model stacking or self-repetition) and lack tunable parameters to tailor its generation breadth. Addition-

ally, neither URLCrazy nor AppCrazy enforce username length or character constraints aligned with OSN policies, which leads to a significant fraction of invalid candidates that cannot be used on real-world platforms like X or Instagram.

Therefore, achieving intermediate performance between UsernameCrazy and these tools would entail designing constrained variants of UsernameCrazy that intentionally reduce the combinatorial space (e.g., by limiting model-stacking depth, filtering out certain insertions, or enforcing stricter edit-distance cutoffs). While such tuning is technically feasible, it would require additional engineering effort to restrict the generation pipeline, which goes beyond the default goal of maximizing discovery potential. Nonetheless, this points to an interesting direction for future work: enabling a flexible knob in UsernameCrazy to balance between comprehensiveness and practical tractability depending on deployment constraints.

Effectiveness of Generation Models. Next, we want to better understand which of UsernameCrazy’s generation models are the most important. We repeat our effectiveness analysis for the *primitive models*, the generation models with *self-repetition* and the *model stacking* of UsernameCrazy for all the usernames of the *Initial Seed*. We compare the number of produced username variants that actually exist on the network or have already been suspended by the platform. We also measure how many active users found are marked as potential bots by Botometer [281] as this is a further indication of the ability of UsernameCrazy to produce relevant usernames. We set Botometer’s detection threshold to 0.95 (see Section 4.4.4). Figure 5.2 summarizes our results. We note that, in the *model-stacking* column we report the outcome when UsernameCrazy applies *exactly two distinct primitive models in sequence* ($A + B$) to every seed handle each time, discarding any output that would exceed the platform’s ‘15-character’ limit. All pairings of our primitive models were evaluated. The three most productive combinations were (i) *Number Insertion* with *Vowel Insertion*, (ii) *Number Insertion* with *Underscore Insertion*, and (iii) *Vowel Insertion* with *Underscore Insertion*.

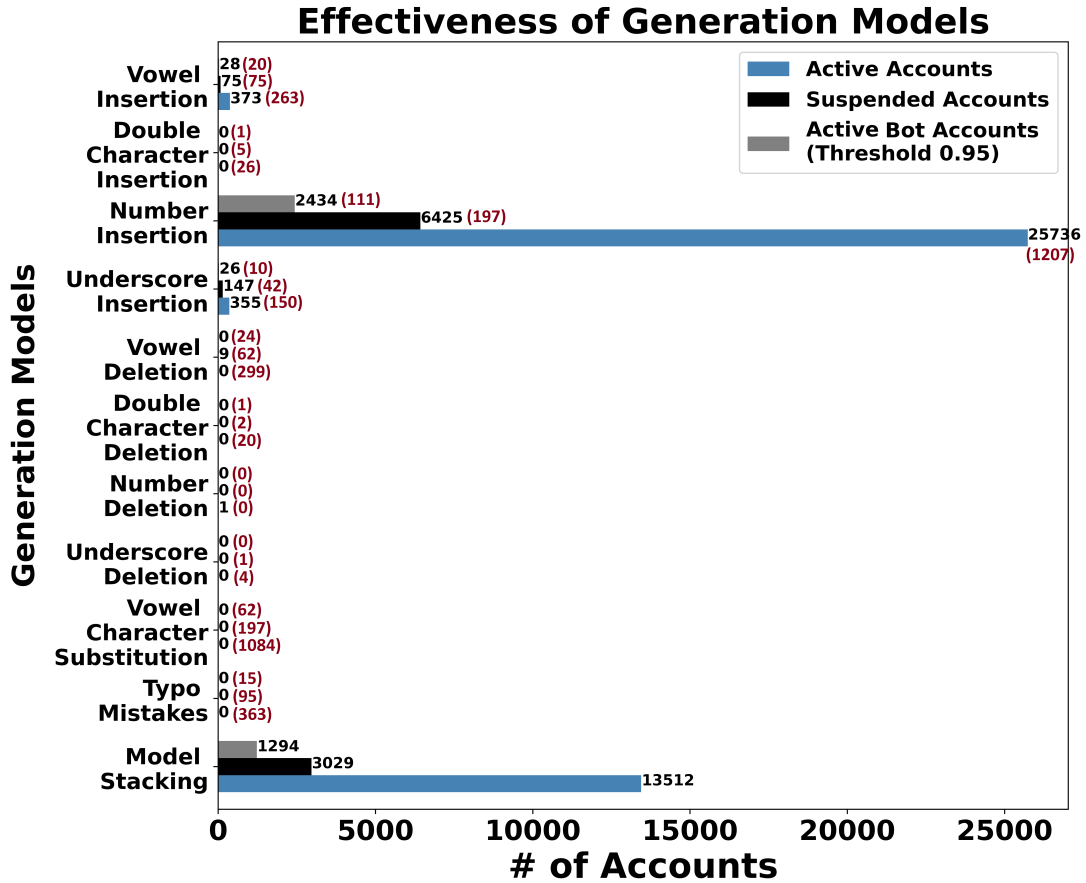


Figure 5.2: The effectiveness of a) each generation model with *self-repetition* and b) *model stacking*. In *parenthesis* is the effectiveness of the primitive models—not shown on the bars.

We observe that a very large number of the generated variants have been either already suspended or declared as potential bots from Botometer. This demonstrates that our username generation models can reveal numerous squatted accounts that are potentially malicious. Moreover, we observe that the number insertion method is the most successful in identifying active accounts and that the majority of such accounts are either suspended or marked as bots. This reveals that a common username squatting tactic on OSNs is to add a digit as a prefix or as a suffix on a popular account’s username. This strategy is simple to both conceptualize and implement. Considering also our measurements on edit distance (Section 4.4.3) we observe that adding between 1–3 digits on a target username seems to be the most common strategy.

UsernameCrazy Runtime Efficiency. Since UsernameCrazy can generate thousands

of candidate usernames, it is important to do that efficiently. To measure UsernameCrazy’s runtime performance, we provided as input to UsernameCrazy the *Initial Seed* (i.e., the usernames of the 97 most popular users for X). We set the maximum character count of a variant to be generated to be 15 (which is also the maximum allowed for X usernames) [363]. UsernameCrazy ran on commodity hardware (a laptop with an Intel Core i5-8265U processor at 1.6 GHz with 8 GB of RAM). UsernameCrazy was able to produce 851,682 valid usernames in 7.6 seconds, which verifies that it can be a practical component in an end-to-end discovery of squatted accounts.

5.4.2 Classification Performance of SQUAD.

Training and Evaluation Metrics. UsernameCrazy produces relevant usernames but not all of them are used with malicious intent. Our classification component aims to take pairs of <seed, squatted> accounts and decide whether the squatted version is suspicious. To evaluate the classification performance of the model we split our dataset to 70% (965 users), 20% (275 users) and 10% (138 users) for training, validation and testing, respectively. Then, we evaluate 7 of the most popular classification algorithms that have been shown to perform well on binary classification tasks: Random Forest [130], Naive Bayes [209], Logistic Regression [61], K-Nearest Neighbor (KNN) [228], Support Machine Vector (SVM) [59], Decision Tree [353], Neural Network (NN) [116]; all using the default parameters⁴.

More specifically we used :

- **Random Forest** [130]: `n_estimators=100, criterion=gini, max_depth=None`.
- **Logistic Regression** [61]: `penalty=l2, C=1.0, solver=lbfgs, max_iter=100`.
- **K-Nearest Neighbors** [228]: `n_neighbors=5, metric=minkowski, p=2`.

⁴Package versions: `scikit-learn` 1.3.0

- **Support Vector Classifier** [59]: `kernel=rbf, C=1.0, gamma=scale`.
- **Gaussian Naive Bayes** [209]: `var_smoothing=10-9, priors=None`.
- **Decision Tree** [353]: `criterion=gini, max_depth=None`.
- **Multi-layer Perceptron (Neural Network)** [116]: `hidden_layer_sizes=(100,), activation=relu, solver=adam, max_iter=200`.

Precision, *Recall* and *F1-Score* [72] have been used as evaluation metrics. We found that Random Forest performed the best across all metrics. To further improve its performance, we fine-tune its parameters on the training set and use the validation set to guide ‘*Hyperparameter Tuning* [98]’ on all parameters, namely ‘*n_estimators, min_samples_split, min_samples_leaf, max_features, max_depth* and *bootstrap*’.

Overall Performance. Random Forest performs better than the other classifiers, with 94% and 94.5% average precision and recall, respectively. Table 5.3 summarizes the classification performance results of all models. While not shown in the table, the accuracy of the Random Forest is 94.44%. The model correctly classifies 154/162 accounts as malicious.

We also compute the ‘*Mean AUC*’ score for all classification models and illustrate the trade-off between the false positive and true positive rates in Figure 5.3. We observe that there is an 99% chance that Random Forest will correctly distinguish negative and positive classes, outperforming the other models. The number of false positives plays a crucial role in eliminating the manual analysis effort for identifying malicious attempts. After a closer inspection we find that our model returns 8 false negative cases, but even though it misses a few cases we already demonstrated that our approach is vastly superior to existing squatting generation techniques while it is more practical than expensive deep graph traversals. We also find 15 false positives which shows that SQUAD keeps the number of false alarms low.

Table 5.3: Classification performance. (P) shows the positive class.

Models	Precision	Recall	F1 - Score
Random Forest			
(P) Suspicious	0.97	0.94	0.95
Benign	0.91	0.95	0.93
Naive Bayes			
(P) Suspicious	0.98	0.82	0.89
Benign	0.78	0.97	0.86
Logistic Regression			
(P) Suspicious	0.94	0.90	0.92
Benign	0.85	0.91	0.88
SVM			
(P) Suspicious	0.95	0.90	0.92
Benign	0.85	0.93	0.89
Decision Trees			
(P) Suspicious	0.94	0.93	0.94
Benign	0.90	0.91	0.90
K-Nearest Neighbor			
(P) Suspicious	0.93	0.86	0.89
Benign	0.81	0.90	0.85
Neural Network			
(P) Suspicious	0.96	0.90	0.93
Benign	0.86	0.94	0.90

Improving SQUAD. SQUAD’s dataset was limited to a relatively small number of manually labelled users. One potential improvement could be to increase the size of the labelled dataset.

We also investigated the false positive (FP) and false negative (FN) cases. For the 15 benign users SQUAD misclassified (FP) we observed that a) all users have small *username* edit distance (≤ 3), b) 7/15 accounts have small *profile name* edit distance (≤ 3), c) 9/15 profiles have similar image with the target and VGGFace2 returns a score lower than the threshold (marked as similar). These explain why SQUAD predicted them as suspicious.

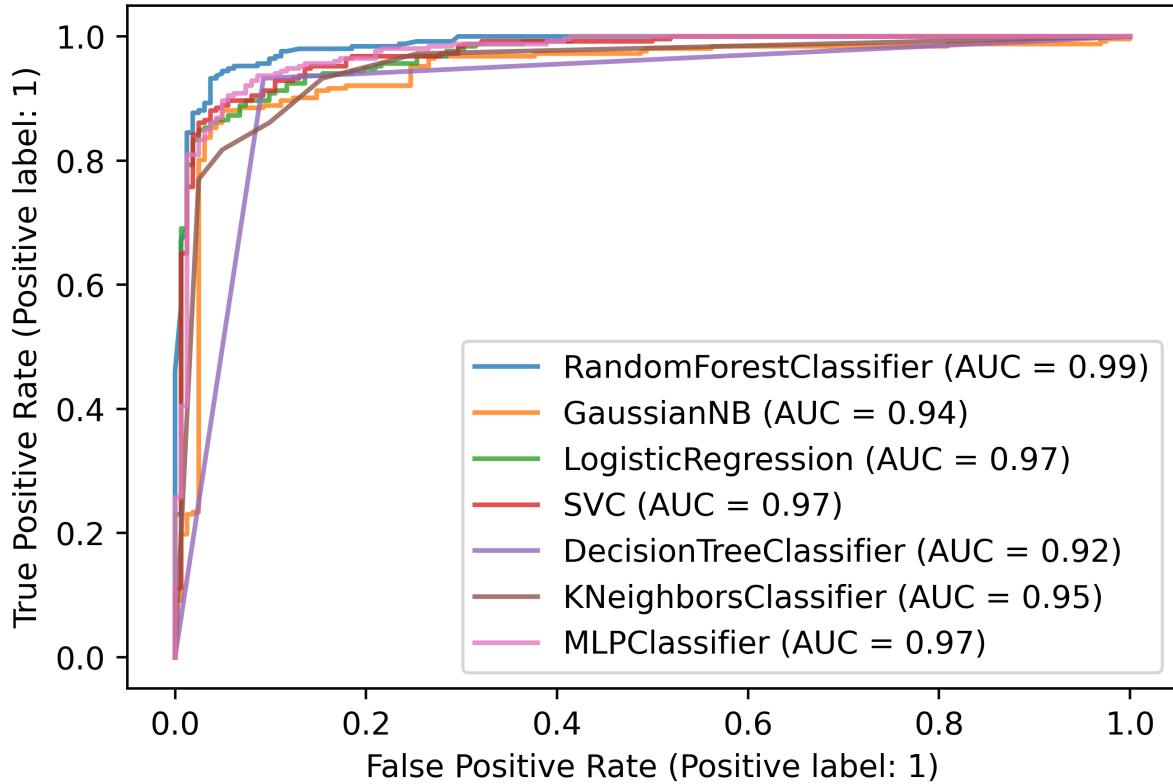


Figure 5.3: ROC Curve of all the classification models.

Looking closer at the latter set, we observed that (d) 6/9 use the word ‘fan’ or ‘parody’ in their bio and 4/6 of these exhibited a high bio similarity (> 0.6). We leveraged these and introduced a post-filter on SQUAD which uses a keyword search for ‘fan’ and ‘parody’ in account bios and negates the suspicious label if it finds a match. This reduced our false positives by almost 30%. We leave integrating more complex methods based on word/sentence embeddings on the bios for future work.

For the 8 malicious accounts SQUAD missed (FN) we made the following observations: a) all accounts have small *username* edit distance (≤ 3), b) half of the accounts have small *profile name* edit distance (≤ 3), c) 6/8 accounts have similar image with the target but VGGFace2 returns a score greater than the threshold and d) 4 of the latter accounts have also very low bio similarity (< 0.10). As a result, the combination of the misclassification of the image and the low bio similarity lead to a wrong classification. Thus, SQUAD can improve as VGGFace2 or image recognition improves—its modular design allows for simply exchanging the face recognition module with another one. Also,

SQUAD only supports image similarity of faces and avatars. Hence, it can miss cases where an impersonator uses a different subject than a face in its profile photo, but as shown in prior work such strategies are less effective [24, 316]. Lastly, SQUAD can also benefit from integrating more advanced models for characterizing users’ bios, activity and tweets.

5.5 Applying SQUAD

In this section, we apply SQUAD to assess impersonation risks across user profiles of varying popularity on social network X . For non-popular accounts, SQUAD flagged no suspicious accounts, with one false negative detected post-verification. In contrast, applying it to celebrity accounts identified that more than half of profiles as suspicious, indicating impersonation intent. Further analysis of flagged accounts revealed URL sharing patterns and ‘follow me’ calls to expand followers. Extending this framework to other platforms, we observe username squatting across social media, with findings on Instagram [145] and TikTok [318] summarized by impersonation levels in Table 5.5.

Applying SQUAD to Non-Popular Accounts. Using the CrowdFlower AI gender predictor dataset [85] we randomly select 15 non-private profiles with less than 200 followers (98% of X users have less than 400 followers [309]). Users without a face in their profile picture are disregarded from the aforementioned selection. For the above 15 non-private profiles SQUAD returns 159 active accounts where 61 have a face in their picture. Out of the latter, 0 (0%) classified as *suspicious* and 61 (100%) as *benign* accounts by SQUAD. After manual verification, we find only 1 *false negative* case. The results contradict the insights of [108], where any account on X can be a victim of impersonation. We summarize our findings in Table 5.4).

Applying SQUAD to Celebrities. We apply SQUAD to 21 users ² of our *Initial Seed* (see Section 4.2.2). SQUAD returns 3,295 active accounts where 1,137 have a face in

Table 5.4: SQUAD results on non-popular accounts.

SQUAD - Non Popular Accounts											
Original Account	Generated Accounts	Accounts with ED 1-3	Active Accounts	Active Accounts with ED 1-3	Suspended Accounts	Suspended Accounts with ED 1-3	Followees	Followers	Classified Malicious by SQUAD	Classified Benign by SQUAD	Active Accounts w/o Face in Profile Image
Account1	11, 876	6, 045 (50.9%)	3 (0.02%)	0 (0%)	0 (0%)	0 (0%)	502	186	0 (0%)	1 (33.3%)	2 (66.6%)
Account2	5, 975	3, 225 (53.9%)	1 (0.01%)	1 (100%)	0 (0%)	0 (0%)	195	15	0 (0%)	0 (0%)	1 (100%)
Account3	12, 929	4, 718 (36.4%)	44 (0.34%)	30 (68.1%)	2 (0.01%)	2 (100%)	33	94	0 (0%)	19 (43.2%)	25 (56.8%)
Account4	6, 983	3, 179 (45.5%)	67 (0.96%)	40 (59.7%)	2 (0.03%)	2 (100%)	333	69	0 (0%)	23 (34.3%)	44 (65.7%)
Account5	11, 612	5, 749 (49.5%)	1 (0.01%)	1 (100%)	0 (0%)	0 (0%)	0	0	0 (0%)	1 (100%)	0 (0%)
Account6	1, 002	916 (91.4%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	128	145	0 (0%)	0 (0%)	0 (0%)
Account7	4, 541	1, 951 (42.9%)	7 (0.15%)	7 (100%)	0 (0%)	0 (0%)	4	4	0 (0%)	3 (42.9%)	4 (57.1%)
Account8	1, 454	1, 287 (88.5%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	173	65	0 (0%)	0 (0%)	0 (0%)
Account9	13, 756	6, 883 (50%)	23 (0.16%)	18 (78.3%)	1 (0.01%)	1 (100%)	0	0	0 (0%)	6 (26.1%)	17 (73.9%)
Account10	3, 116	2, 860 (91.8%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	121	129	#FP = 0	#FN = 1	0 (0%)
Account11	4, 159	2, 854 (68.6%)	3 (0.07)	3 (100%)	0 (0%)	0 (0%)	79	140	0 (0%)	1 (33.3%)	2 (66.6%)
Account12	1, 779	1, 644 (92.4%)	2 (0.11%)	2 (100%)	0 (0%)	0 (0%)	14	6	0 (0%)	2 (100%)	0 (0%)
Account13	8, 164	3, 885 (47.6%)	7 (0.09%)	5 (71.4%)	0 (0%)	0 (0%)	490	119	0 (0%)	4 (57.1%)	3 (42.9%)
Account14	6, 360	4, 627 (72.8%)	0 (0%)	0 (0%)	0 (0%)	0 (0%)	78	28	0 (0%)	0 (0%)	0 (0%)
Account15	7, 883	4, 524 (57.4%)	1 (0.01%)	0 (0%)	0 (0%)	0 (0%)	62	38	0 (0%)	1 (100%)	0 (0%)
Total	101, 589	54, 347 (53.5%)	159 (15.7%)	107 (67.3%)	5 (4.92%)	5 (100%)	2, 212	1, 038	#FP = 0	#FN = 1	98 (61.6%)

their profile picture. Out of the latter, 625 (55%) classified as *suspicious* and 512 (45%) as *benign* accounts by SQUAD. Alarmingly, 297 (47.5%) users from the suspicious set were generated by the *number insertion* strategy, signifying the model’s malicious squatting intent (see Section 5.4.1).

5.5.1 Discussion

Analysing the Behavior of Suspicious Profiles. To identify whether the suspicious variants are engaged in malicious activities, we analysed the content of their tweets. Using the full-archive search endpoint (API v2 [361], see Section 4.3.2) we fetch their 500 most recent actual tweets, resulting in 8,516 tweets where 1,476 (17.3%) include a URL and 5,056 (59.3%) are *English* tweets. We manually observed two trends in the tweets of malicious accounts which we further measured. Firstly, they share URLs which sometimes look suspicious and secondly they try to grow their follower base. In terms of URLs in tweets, we first analyze their integrity. Since none of the URLs are present in the VirusTotal database [304] we submit them for scanning. Alarmingly, we discover: i) 24 (1.63%) URLs were marked as malicious by 1-3 antivirus systems, ii) 533 (36.1%) URLs fail to use *HTTPS*, and iii) 16 (1.08%) URLs automatically obtained a TLS certificate from *Let’s Encrypt* Certificate Authority [182], a service that has faced criticism for its decision to abstain from implementing any form of security checks prior to granting certificates to domain owners [6]. We also measured follower base growth attempts in the English

tweets using a simple keyword search approach, looking for the phrase *follow me* (with slight variations acceptable). We found 266 (5.3%) such cases, being posted by 78 distinct users.

Username Squatting across Platforms. Username squatting can be exploited on other OSNs that use usernames as unique identifiers to search for and interact with users, such as Instagram and TikTok. To demonstrate the extent of the problem on other platforms we would ideally have to (a) adapt UsernameCrazy generation models to the username constraints of each platform, (b) collect existing variants on each platform and, (c) classify each account. The former is easy to do and can be efficiently executed. (b) on the other hand is harder especially since there are no publicly available APIs to allow us to automatically crawl Instagram and TikTok. Hence, this would require developing specialized web crawlers for each case. Lastly, (c) is also hard as it would entail manual analysis on either all or a relatively large subset of the existing variants. Instead, we design a more targeted experiment which can quickly allow us to collect preliminary indicators of the extent of the problem on other OSNs.

Furthermore, we observe that a lot of popular profiles maintain accounts on other OSNs with the same username. Therefore, if other OSNs are plagued with similar issues, then it is likely that variants that were marked as malicious on *X* could appear on these OSNs too. To examine this, we select the top 10 accounts of our *Initial Seed* and manually check how many exist with the same username across both Instagram and TikTok; this resulted in 5 profiles. For each profile we use the squatted usernames, previously generated by UsernameCrazy, and find how many still exist on *X*. Then, for every active variant we manually select the ones that are likely impersonators and examine whether they a) exist with the same username, b) have a similar image and c) perform potential malicious activity on other OSNs. We call ‘*Level 1*’ the users who cover only the first property, ‘*Level 2*’ the profiles which cover the first property plus any of the other two and ‘*Level 3*’ the accounts that cover all the aforementioned properties.

Table 5.5 summarises our findings. We find that the most common pattern of the active

squatted accounts across all OSNs remains the *Number Insertion* pattern. More importantly, we observe that almost half of the squatted accounts that impersonate on *X* exist on other OSNs and have common properties.

Table 5.5: Similar impersonation attempts across various Online Social Networks.

Impersonations Across OSNs				
Users	<i>X</i> - Active Impersonations	Category	Instagram	TikTok
@katyperry	7	Level 3	1/7	0/7
		Level 2	4/7	4/7
@justinbieber	20	Level 3	3/20	4/20
		Level 2	7/20	10/20
@cristiano	1	Level 3	0/1	0/1
		Level 2	1/1	1/1
@ladygaga	10	Level 3	0/10	1/10
		Level 2	3/10	7/10
@kimkardashian	5	Level 3	2/5	0/5
		Level 2	3/5	0/5
Total	42	Level 3	6/42	5/42
		Level 2	18/42	22/42
		Level 1	27/42	28/42

5.5.2 Responsible Disclosure.

We believe it should be the responsibility of the platform to integrate automated processes to identify impersonators and suspicious accounts. We reported all the manually verified *impersonators* (see Section 4.4.4) to Twitter via a formal report at HackerOne program [121]. Twitter acknowledged the problem and marked our report as *Informative*. Unfortunately, Twitter’s response [180] suggests that the platform simply relies on weak crowdsourcing methods [323] for identifying impersonation attempts even though it is aware of the problem. Such methods are known to suffer from erroneous reports and manual verification of crowdsourced reports does not scale.

5.5.3 Code Availability

The implementation of the end-to-end framework (SQUAD) described in this chapter is available at a GitHub repository [173] under the MIT open-source license available at GitHub [174]. For instructions on how to run the code in order to reproduce the results of this Chapter refer to ‘README.md’ file.

5.6 Conclusion

Our study is the first to characterize *username squatting* on OSNs. We showed that *username squatting* can be used for confusion and impersonation. We found hundreds of thousands of *typo-mentions* mentioning squatted versions of popular accounts with no apparent mentioner–mentionees relationship, and demonstrated how the search recommendation algorithm of a popular OSN can unwittingly amplify online confusion. We discovered that thousands of squatted usernames are already suspended by X while tens of thousands of *active* squatted accounts are likely bots. While not all username variants are malicious, we show that *username squatting* can be a useful signal for identifying such accounts. We designed SQUAD, to aid detection of malicious accounts on OSNs. SQUAD incorporates new techniques for username squatting along with a similarity feature extractor and classification components. SQUAD can generate tens of thousands of relevant squatted usernames of a given account in a few seconds, and can achieve 94% F1-score in detecting suspicious squatted accounts when trained on a small dataset. SQUAD can be used by account owners to identify attempts to impersonate them, by regulating authorities such as the Federal Trade Commission (FTC, US) [10] or the Information Commissioner’s Office (ICO, UK) [137], and the OSN platforms themselves who want to take measures to limit the negative impact impersonation and confusion can have on the community. Lastly, we revealed that username squatting targets primarily popular accounts and 36% of the suspicious accounts’ tweets share insecure URLs and 5% of the

tweets try to grow their or others' follower base.

Chapter 6

Attribute Leakage in Text-To-Image Models

While the previous chapters focused on explicit identity impersonation via squatted usernames on social platforms, this chapter shifts attention to implicit identity inference—a subtler but equally critical threat to online identities in the era of generative AI. Both cases illustrate a broader trajectory: adversaries leveraging accessible digital outputs—whether usernames or generated media—to extract, infer, or manipulate identifying information. In contrast to conventional identifiers like usernames, the threats examined here arise from latent signals embedded within synthetic content, underscoring how modern identity risks extend beyond user-chosen labels to encompass attributes that individuals may not even realize they are exposing.

More specifically, this Chapter, presents a detailed investigation into attribute leakage in text-to-image models, focusing on the vulnerabilities related to authorship and sensitive attribute inference. It presents a novel framework for inferring properties through black-box adversary models that leverage input prompts, generated outputs or a combination

of both. Moreover, the models are thoroughly evaluated across various text-to-image configurations. The Chapter also examines several state-of-the-art defense mechanisms, assessing their robustness against the framework, while highlighting critical gaps.

6.1 Problem Statement

Diffusion models are part of the generative artificial intelligence paradigm and have revolutionized the task of image synthesis. By adding noise to the training data and learning how to reverse the process, they can create images that are indistinguishable from real images to the naked eye [269]. Diffusion models have successfully conditioned the image generation process based on *natural language instructions* (**NLIs**) which enabled radical ease of use by non-sophisticated users. Users can simply describe what they want the models to generate using natural language and the underlying *text-to-image* (**T2I**) model will generate an image matching the description. Several T2I models are widely accessible such as *Stable Diffusion*, *Latent Diffusion*, and *DALL-E* among others and millions of users are leveraging them to create tens of millions of images [184].

The increasing prevalence and popularity of those models comes with privacy risks. Such risks have not been studied to date and this work aims to take the first rigorous step towards developing this understanding. To better illustrate the potential privacy leakage in T2I models let us consider the following scenario which is also depicted in Figure 6.1. Alice and Bob are concerned about recent political events. They want to share their support or criticism on online forums and networks but they are afraid of potential targeting or backlash. Alice and Bob leverage speech or text to anonymously instruct a T2I model to generate a related image. This can happen by executing an open source model offline (e.g. Stable Diffusion) or by using online tools without a registered account (e.g. craiyon). The ultimate goal of the T2I model is to achieve the best possible alignment of the generated image with the given instruction in a common latent embedding space. The textual descriptions (NLIs) and the generated images are by default known to the

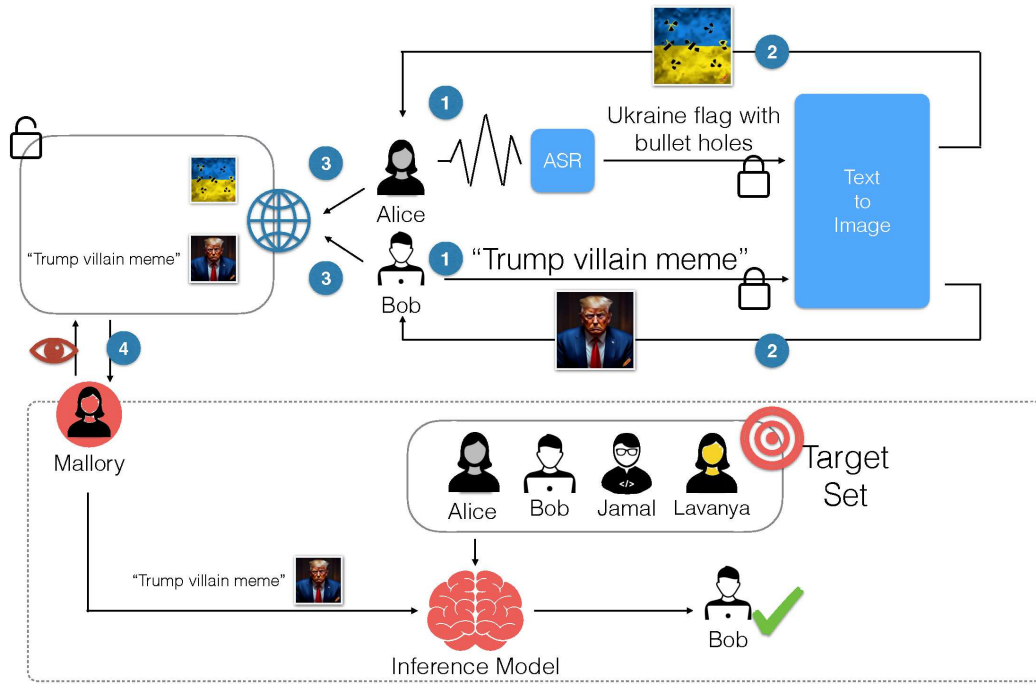


Figure 6.1: Privacy leakage in Text-to-Image model usage. Users create natural language instructions which are used to guide the generation of images. Users share their instructions and/or the generated images with a false expectation of privacy online and by default with online model providers. Mallory who observes the shared data, aims to infer sensitive attributes about the data owner. This Figure depicts the scenario of authorship inference. Disclaimer: the images at Step 2 were generated solely for the purpose of illustration using *craiyon* [62] which at the time of writing was available for free online and anonymous usage.

model provider. Alice and Bob then anonymously share their generated content on a public forum [184].

Mallory has access to publicly shared data. If Mallory can infer the author of an NLI, this poses serious privacy risks. Authorship leakage can compromise user privacy, especially when anonymity is essential. For example, in some countries, online political content is heavily censored, with artists facing prosecution or threats [102, 22]. Dissident artists may rely on T2I-generated images to express their views anonymously. Similarly, online discussion groups frequently use AI-generated memes to critique political ideologies [337]. To minimize potential backlash or draw less attention to themselves, some users appear to participate using aliases or pseudonyms.

To get a clearer picture of the pervasiveness and nature of anonymity in such settings,

we conducted a supporting study on the official *Stable Diffusion* [76], Midjourney [222], DALL-E [67] and DALL-E 2 [4] subreddits. Using the free version of the publicly available scraping tool [306] from Apify [15] we could extract up to 1000 active members of a subreddit. Inspired by Marcondes et al. [211] we leverage a multi-faceted methodology—encompassing pattern recognition, comparison against known name databases (e.g., NLTK names corpus [315]), structural name-like analysis, fuzzy string matching and an additional layer of lexical checks using the WordNet database [225]—we differentiate aliases from real names. For example, this approach flagged user handles containing recognisable nouns like ‘shark’ or ‘container’ as more likely to be aliases. To assess the performance of this methodology, we first tested it on the Stable Diffusion subreddit. Of the 653 unique usernames scraped, 648 were classified as aliases and only 5 as real names. Then, an independent rater performed a manual analysis to these usernames, resulting in 619/29/3/2/95.5/99.7 for TP/FP/TN/FN/Precision/Recall. We follow the same procedure for the rest of the aforementioned subreddits. Table 6.1 shows our findings, highlighting only few misclassifications, suggesting that a number of online users choose to use aliases, potentially as a means of reducing personal identifiability.

Table 6.1: The number of unique active members that use aliases or real names as usernames in Stable Diffusion, Midjourney, DALL-E and DALL-E 2 subreddits.

Unique Active Members (First 1,000 results per subreddit)				
	Subreddits			
Usernames	Stable Diffusion	Midjourney	DALL-E	DALL-E 2
Aliases	648	738	267	235
Real Names	5	2	1	1

There is no expectation that T2I usage (providing an NLI input) should reveal personal attributes of the user.

Under the EU GDPR and its UK counterpart, however, ‘personal data’ encompasses any information relating to an identifiable person, including ‘inferences’ or ‘profiles’ generated from otherwise non-identifying inputs (Art.~ 4(1) and Recital~30) [326] [89]. Likewise,

the California Consumer Privacy Act (CCPA) defines personal information to include ‘inferences drawn ... to create a profile about a consumer reflecting the consumer’s preferences, ...’ (Cal. Civ. Code 1798.140(v)(1)(K)) [1]. Legal scholarship confirms that inferred data can trigger full data-protection obligations—lawful basis, transparency, and data-subject rights—even when the original input appears innocuous [335] [332]. Consequently, if an observer can derive a property such as authorship or demographic traits from a user’s T2I prompt, the practice may conflict with these regulatory regimes and undermine user expectations of privacy. Potential authorship inferences therefore represent a concrete privacy risk rather than a merely theoretical concern.

On the other hand, T2I models have been purposefully exploited by malicious actors, to launch misinformation campaigns or distribute toxic or unsafe content online [283, 253]. Such content which is illicit in nature is also distributed with an expectation of anonymity. Potential authorship inferences can facilitate attribution of unsafe content to their source. This a desirable orthogonal approach to safety filters that can be employed responsibly by model providers. Safety filters can be disabled or bypassed [371, 20] and authorship attribution can help regulators and enforcement authorities identify and prosecute culprits.

To the best of our knowledge, attribute leakage in text-guided diffusion-based image generation has not been studied. This work aims to bridge that gap. Previous research has explored authorship inference in other domains, such as book writing and Wikipedia article contributions [302, 305]. However, these studies focus solely on text, whereas T2I models may leak sensitive information from both the input text and the generated images.

Some inferences from images are proven possible [219, 27]. Existing works however, do not study AI-generated images and thus cannot be used for authorship inferences or for studying the relationship of the leaked attribute with guiding NLIs. Attribute leakage in T2I is also more challenging as the models use a stochastic process in their generation and thus text input information might be lost. In addition and uniquely to T2I models, leakage can be multi-modal as inferences can be potentially performed by combining both

the input text and the output image. The only existing work on T2I attribution focuses on identifying the source T2I model in a closed-world setting with four target models [290]. Our study instead investigates the potential for *user* attribute leakage. Furthermore, we examine an understudied attribute that affects a vulnerable population.

Our Approach. In this work, we address these challenges and investigate the novel research question: *Do text-guided diffusion models for image generation leak user attributes?* Motivated by our guiding examples, we conduct a thorough analysis of authorship leakage through a set of targeted attacks, which we term *spear authorship inference* attacks. To further demonstrate attribute leakage, we examine pathological language characteristics associated with dementia, a neurocognitive condition, and show that text-guided diffusion models can encode such highly sensitive traits.

To answer our overarching question we design an attack strategy based on a hierarchical black-box adversary. The adversary has access to different levels of information and can succeed through black-box only observations of T2I usage, making our threat model both more realistic and stronger compared to a white-box alternative. Specifically, our adversary can employ three inference strategies: *Input Inference* ($\mathcal{A}_i$) who only has access to the input natural language instruction; *Output Inference* ($\mathcal{A}_o$) who only has access to the output image of the model; and *Multi-Modal Input-Output Inference* ($\mathcal{A}_{io}$) which has access to both the natural language input and the image output of the T2I model.

Our approach to mitigate the effects of the probabilistic generation process is based on two key insights. First, T2I models utilize embedding models (e.g., CLIP [254]) that learn a multi-modal embedding space by jointly training on images and their textual descriptions. Extracting image embeddings using this approach should produce representations closely aligned with their NLIs, increasing the likelihood of retaining input information. Second, T2I models’ denoising process allows for creative freedom and unless explicitly controlled, the same NLI can result in very different images every time is used. We leverage this insight and augment our image training set and show that this helps generalize our models

to not only improve performance but also achieve highly accurate inferences from images generated from T2I models unknown during training.

Our extensive evaluation confirms that *spear authorship inferences* are feasible from both the input and output of T2I models, with text-image alignment embeddings significantly improving inference performance. Additionally, our models remain robust in both closed and open-world settings and against state-of-the-art obfuscation techniques. Finally, we show that pathological language markers present in input text persist through the diffusion process via text-image embeddings, highlighting the risk of leaking sensitive neurocognitive conditions, such as dementia, from generated images.

Contributions. We summarize our key contributions as follows:

- **Novel Application Domain.** To the best of our knowledge, we are the first to study attribute leakage in T2I models. We hope our work inspires further research on T2I leakage of novel attributes and the development of effective mitigation strategies.
- **New T2I Black-Box Inference Method.** We propose a novel framework for inferring user attributes in T2I model usage through hierarchical black-box access, leveraging Text-based, Image-based, and Multi-Modal Inference Models built on text-image alignment embeddings.
- **New Findings.** We demonstrate that T2I images leak authorship and can encode dementia language markers, enabling dementia inferences from both text input and generated images. Our best multi-modal classifier achieved over 79% accuracy on the ADRess dataset.
- **Rigorous Evaluation.** We conduct a rigorous evaluation, demonstrating that our attacks achieve state-of-the-art performance even with limited training samples. Our approach remains robust against unseen T2I models and state-of-the-art mitigation strategies.

The rest of the Chapter is organized as follows. Section 6.2 describes the formulation and implementation of our adversarial models. Section 6.3 elaborates on our experimental setup, dataset, metrics and research questions. Section 6.4 presents the results and analysis of our evaluation. Section 6.4.4 discusses the applicability of the attack to a novel sensitive attribute (dementia). We conclude the Chapter in Section 6.5.4.

6.2 Black-Box Attribute Inference Attacks

6.2.1 Threat Model and Framework Overview

Threat Model. Users of text-to-image models frequently share either their prompts, or the corresponding generated images, or both online. For example, Lexica website [184] contains a large number of generated images from Stable Diffusion and the corresponding prompts for studying prompt engineering and AI-generated images. Lexica also provides an API which allows anyone to retrieve either prompts or images based on provided criteria. Users also share such information on online communities such as the “Unstable Diffusion” community which distributes unsafe images generated with Stable Diffusion. There is a false assumption that sharing such data can be done anonymously, as there is no expectation that neither prompts, and especially nor the synthetic images can leak authorship information. In our work, we investigate the following overarching questions: *Do text-to-image model inputs contain identifiable characteristics of the prompt’s author? Do these characteristics transfer to the generated outputs? And can an adversary with access to raw prompts or images infer the author’s identity with some probability?*

Our threat model is illustrated in Figure 6.2. More formally, we consider an adversary $\mathcal{A}$ with black-box access to text-to-image models. Black-box attacks pose a greater challenge than white-box attacks, as the adversary lacks access to the model’s internal structure or computations. However, this approach is more realistic, has broader implications, and is harder to defend against, as it can be executed not only by T2I model providers but by

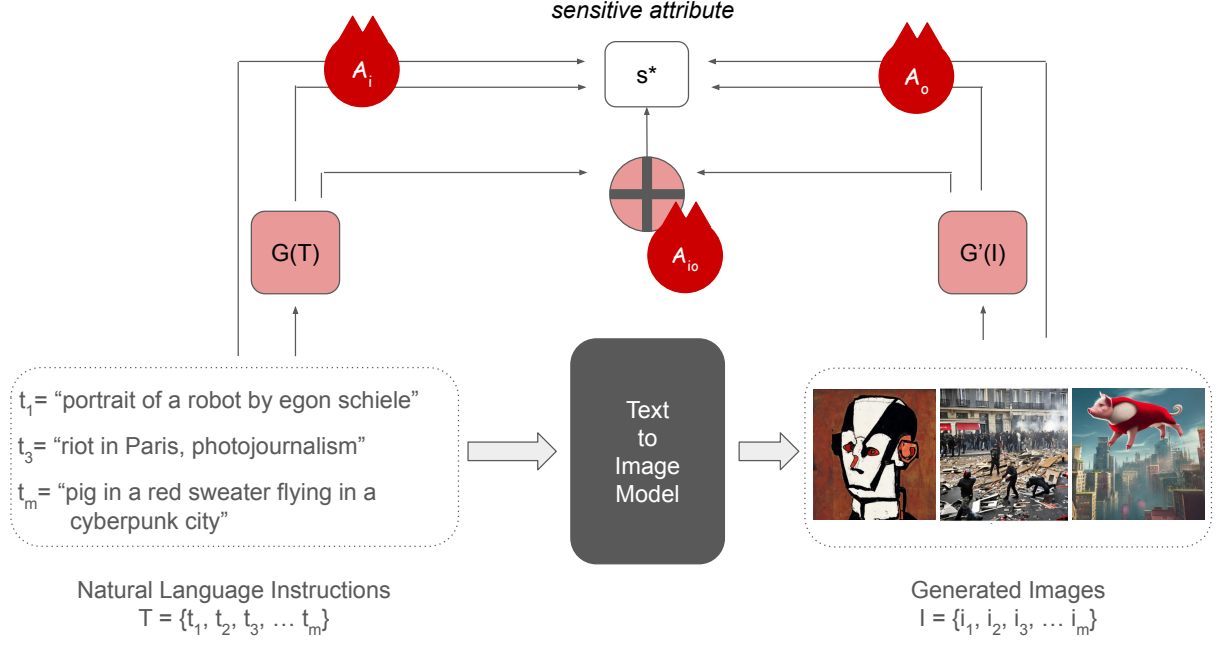


Figure 6.2: High-level taxonomy of attribute (s^*) inference attacks against text-to-image models. We assume a hierarchical adversary with access to different information of the text-to-image model: (1) $\mathcal{A}_i$ infers the attribute from the input prompts P only, (2) $\mathcal{A}_o$ infers the attribute from the output images I and (3) $\mathcal{A}_{io}$ combines both the input and the output of the target model for inference.

anyone who can observe the model's inputs and/or outputs.

To formulate the adversary's goal, we adopt the general definition of attribute inference attacks outlined by Song et al. [302]. We assume that the adversary $\mathcal{A}$ has access to an auxiliary dataset D_{aux} containing labelled data in the form of (x, s) , where x belongs to the input domain X ($x \in X$) and s represents a discrete attribute ($s \in S$), with S being the set of all possible attribute classes. The goal of $\mathcal{A}$ is to infer a sensitive attribute $s^* \in S$ given a seemingly innocuous and previously unseen input $x^* \in X$. To achieve this, the adversary aims to learn an adversarial function f on D_{aux} that maps the input domain X to the discrete attribute space S . Formally, this is defined as $f_\theta : X \rightarrow S$, where θ represents the learnable parameters of the function f . Furthermore, we consider an extended scenario where the adversary can infer authorship not only from the raw input but also from an embedding function G , such that $g_\phi : G(X) \rightarrow S$, where ϕ are the learnable parameters of G .

For clarity of presentation, we will first focus our analysis on *spear authorship inferences*. In Section 5.5.1, we will extend this analysis to sensitive pathological language markers, demonstrating how such markers can also be encoded in both the input and output of T2I models. More concretely, a *spear authorship inference* adversary $\mathcal{A}$ aims to identify, with some probability, the author $\alpha \in A$ from a finite set of target authors A , where $|A| \leq c$ and $c \in \mathbb{Z}$ is a small constant. In our evaluation, we experimented with values of c ranging from 100 to 200. Although these attacks are less scalable, they can have more severe consequences. Large-scale inference typically targets the general population, leading to broad but diluted effects, such as user profiling for targeted advertising. In contrast, spear authorship inference attacks focus on a small set of high-value individuals—such as company executives or government officials—where the intended harm can extend across entire organizations or societies.

We consider a hierarchical adversary with access to different modalities, both before and after processing (see Figure 6.2 for an overview of our adversarial setup). In particular, an adversary targeting T2I models can have access to natural language instructions (input of the model), access to the generated images (output of the model), or both. In each case, the modalities considered by the adversary are different and thus different models need to be designed accordingly. In the first case the adversarial models need to be designed to work well with text, and in the second case with images, The last case is even more complicated as it requires a multi-modal approach. For a multi-modal approach, the choice of text and image embeddings or features play an important role in the effectiveness of multi-class classification. Next we elaborate on the formulations of each adversary.

Inference Framework Overview. Figure 6.3 illustrates our method for authorship inference in text-to-image (T2I) model usage. Without loss of generality, we assume the target sensitive attribute is authorship, where an author is the user who provides the natural language instruction (NLI) to the T2I model. Binary attribute inference is a special case of this framework.

Using supervised learning, the adversary trains classifiers to infer authorship. Specifically, for a set of users $U = U_1, U_2 \dots U_c$, NLIs $T = T_1, T_2 \dots T_m$, and images $I = I_1 \dots I_m$, where $m \geq c$, they can train:

- An **input-only model** (A_i) using (U, T) pairs.
- An **output-only model** (A_o) using (U, I) pairs.
- A **multi-modal model** (A_{io}) using (U, T, I) triplets.

This approach enables authorship inference across different modalities, improving classification accuracy and robustness.

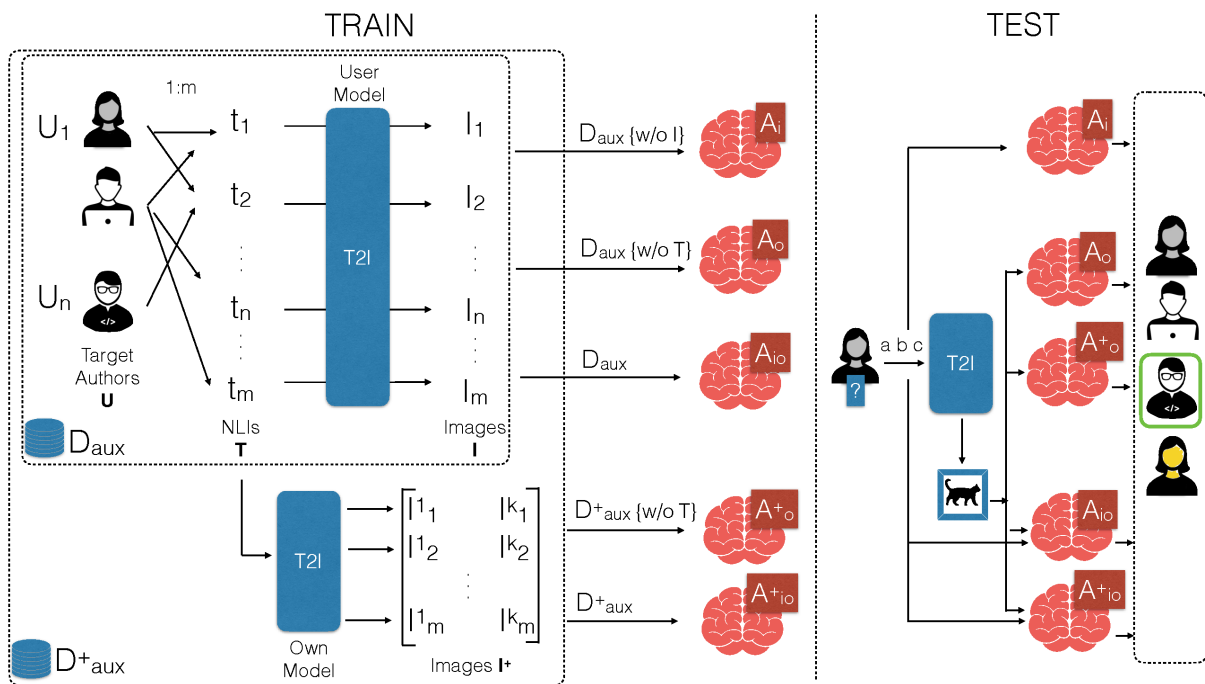


Figure 6.3: Inference Framework Overview.

As we will demonstrate, for authorship inference such models perform very well when the adversary knows the T2I model that is being used, a reasonable assumption given prior model attribution methods [290]. However, in some cases, the adversary may be unable to reliably identify the user's model. Training on images from a different model may not generalize well, limiting inference accuracy.

To address these challenges, we employ data augmentation for training inference models. Specifically, the adversary generates an augmented image set I^+ containing $m \times k$ images, where each $t_i \in T$ produces k new images. By training on $I \cup I^+$, the adversary enhances output-only ($\mathcal{A}_o^+$) and input-output ($\mathcal{A}_{io}^+$) adversarial models.

During inference, the adversary selects the appropriate model to attribute intercepted samples to target users. While augmentation increases computational costs, it improves model transferability and performance (see Section 6.4.3). Additionally, the adversary can readily leverage public T2I models to augment their datasets. Next, we describe the spear authorship inference models. Thus far, we have focused on a *closed-world* setting, where all inference samples originate from a predefined set of target authors. In reality, in several settings such as open discussion forums and online social networks, it is unrealistic to assume the adversary knows and has training data from all authors. Our method can easily be adapted to an *open-world* setting where the inference sample can come from both known and unknown users. We briefly discuss this adaptation in Section 7.1.1.

Challenges in constructing (U, T, I) corpora. Although our experiments draw on DIFFUSIONDB [342]—which already contains *user* U , *prompt* T , and *image* I triples—a targeted scrape of the official Stable Diffusion Discord, these resources illustrate effort, not impossibility. Outside such purpose-built datasets, public T2I artifacts rarely include clean author labels; when it does (e.g., via Discord handles), identifiers are pseudonymous, platform-specific, and class-imbalanced. Consequently an adversary aiming to target an *arbitrary* set of individuals need to : (i) resolve user identities across multiple platforms, (ii) de-duplicate reposts and memes that break the one-to-one (U, T, I) mapping, and (iii) gather a sufficient number of samples per user for a reliable classifier (see our classifier’s performance with 70 training samples per author in Section 6.4.1). These steps are feasible using standard crawlers and weak-label heuristics, but they do demand sustained collection time and careful navigation of rate limits, privacy rules, and ethical approvals, as documented in a prior large-scale generative-model dataset [285].

Rationale for inference when the attacker is the model provider. Even a text-to-image service that stores ground-truth account information may still benefit from an automated authorship-inference pipeline:

- ***Access-control silos.*** In production systems, customer data are deliberately stored in separate, locked-down databases—for example, one shard per region, one instance per large enterprise, or behind differential-privacy guards. As a result, abuse-response or trust-and-safety teams often have log access only to prompts or only to images, not to the full join of (U, T, I) ; an internal inference model bridges that gap without violating data-minimisation principles mandated by GDPR (Article 5(1)(c)) [89] and CCPA (Cal. Civ. Code § 1798.100(a)(3)) [1].
- ***Downstream attribution.*** Content moderators on external platforms (e.g. X or Reddit platform) receive images stripped of provenance tags. To trace a harmful or policy-violating image back to an originating account, the platform—or the provider acting on a legal request—must rely on black-box authorship inference [290] [230].
- ***Forensics and accountability.*** When a user denies having issued a particular prompt (e.g., in contractual or legal disputes), the provider can present a probabilistic attribution backed by an independently trained classifier rather than raw database entries that may be contested or subject to discovery constraints.

Therefore, even with access to ground-truth logs, a model provider has compelling operational, legal, and privacy reasons to pursue the black-box inference capability examined in this work.

6.2.2 Input Attack ($\mathcal{A}_i$)

Formulation. In this setup, D_{aux} consists solely of natural language instructions submitted to a text-to-image model, along with their corresponding author labels: $D_{aux} =$

$\{(t_i, \alpha_i)\}_i$, where $t \in T$, T is the set of all instructions, $i = 1..n$, and n is the total number of training samples. $\mathcal{A}_i$ aims to learn $f_\theta : T \rightarrow S$ and $g_\phi : G(T) \rightarrow S$.

Design and Implementation.

We construct multiple adversarial models to infer authorship from text. Our embedding-based classifiers (g_ϕ) consist of an embedding extraction component followed by a linear classification layer. We experiment with pre-trained InferSent and BERT models for embedding extraction, both of which have demonstrated strong performance in various NLP tasks. Additionally, we fine-tune BERT on our dataset of NLIs. BERT embeddings (a 768-dimensional vector) are fed into a linear layer for multi-class classification, while InferSent embeddings are processed through a dense layer with a softmax activation function. Training is optimized using the Adam optimizer [162].

Our raw-text classifier (f_θ) directly learns from T . We hypothesize that T2I users have distinct interests, subjects, or styles, which influence the vocabulary used in their NLIs. Thus, features like Bag-of-Words (BoW) should effectively capture author differences in instructions. Unlike other stylography features, which are important for large document inferences, BoW focuses solely on the presence of words, not their order or sentence structure. We use BoW to extract features relevant to word occurrence within NLIs, employing a TF-IDF vectorizer. These BoW features are then passed through a dense layer with a softmax activation function for multi-class classification.

Our main $\mathcal{A}_i$ pseudocode is described in Algorithm 3.

6.2.3 Output Attack ($\mathcal{A}_o$)

Formulation. In this setup, D_{aux} consists of image outputs from a text-to-image model along with their corresponding author labels (i, α) , where $i \in I$ and I is the set of all generated images. $\mathcal{A}_o$ aims to learn $g_\phi : G(I) \rightarrow S$. Note that for $\mathcal{A}_o$, we do not consider learning directly from the raw data ($f_\theta : I \rightarrow S$) because we expect abstract attributes

Algorithm 3 Input Attack Algorithm ($\mathcal{A}_i$)**Input** A subset of $U \rightarrow U'$, $t_U \rightarrow T'_U$ and $T_{U_x} \rightarrow T'_{U_x}$ **Output** Top-5 predictions of U'

```

1: procedure AUTHORSHIP_INFERENCE_INPUT_ATTACK
2:    $X[] \leftarrow$  All the prompts of  $T'_U$ 
3:    $y[] \leftarrow$  All the authors of  $U'$ 
4:    $X_{train}, X_{test}, y_{train}, y_{test} = \text{train\_test\_split}(X, y, t\_size)$ 
5:   for model in models do
6:                                      $\triangleright$  For embedding models:
7:      $Embeddings[] \leftarrow$  Extract sentence embeddings
8:      $model \leftarrow$  Train a linear classifier
9:                                      $\triangleright$  For baseline models:
10:     $model \leftarrow$  Train on raw sentences without access to the embeddings
11:    Return Top – 5 Accucary

```

like authorship, or semantically meaningful context related to authorship, to be more effectively captured in image embeddings. These embeddings are generally more useful than handcrafted features extracted from raw data.

Design and Implementation. A straightforward approach to infer authorship from image embeddings is to apply pre-trained CNN models, such as ResNet [125], InceptionV3 [311], or VGG19 [155], to extract image features. These embeddings have proven effective for image classification. CNNs employ a hierarchical structure, enabling more abstract representations compared to handcrafted feature extraction methods like SIFT [199] and HOG [66].

However, our key insight is that inferring *authorship* from images generated by NLI requires embeddings suited for semantic image processing—ones that encode relationships between textual descriptions and images. Traditional image embeddings, while superior to handcrafted methods, primarily capture objects in images but struggle to generalize to unseen objects, styles, and topics. To address this, we use models ($G(I)$) that learn visual representations from natural language supervision. Specifically, our $\mathcal{A}_o$ attack models leverage vision transformer architectures such as CLIP [241] and ViT [352]. CLIP is used in some T2I models to encode NLIs and guide the diffusion process, whereas ViT serves as a more general vision transformer model.

The transformer-based vision embeddings are then passed to various popular classifiers

configured for multi-class classification using the scikit-learn library [248]. In particular, we employ an *SVM* classifier with a *One-Vs-One* approach, a *Logistic Regression* classifier using the *One-Vs-The-Rest* method, and a Naive Bayes classifier leveraging a multinomial model to compute class probabilities. Additionally, we use Decision Trees, Random Forests, Neural Networks, and KNN classifiers with a *multi-label* approach.

Note that training on a single image generated from an NLI for a given T2I model may not always be effective. This is due to the approximate nature of the diffusion process, which can produce different images each time based on user-specified input parameters. For instance, the `cfg` scale parameter (classifier-free guidance scale) in Stable Diffusion controls the diversity of the generation process, determining how strictly the model follows the given instruction. Furthermore, for each NLI and diversity setting, multiple images may serve as representative visual interpretations of the instruction. A key insight of our output attack strategy is to incorporate data augmentation into the training process to address this variability. Specifically, for each NLI instruction, we generate multiple images and use all of them to train the adversarial models. We denote this attack model as $\mathcal{A}_o^+$.

Our main $\mathcal{A}_o$ pseudocode is listed on Algorithm 4.

Algorithm 4 Output Attack Algorithm ($\mathcal{A}_o$)

Input A subset of $U \rightarrow U'$ and $I \rightarrow I'$

Output Top-5 predictions of U'

```

1: procedure AUTHORSHIP_INFERENCE_OUTPUT_ATTACK
2:    $X// \leftarrow$  All the image embeddings of  $I'$ 
3:    $y// \leftarrow$  All the authors of  $U'$  ▷ For each author in  $U'$  extract the image embeddings of  $I'$ 
4:    $X_{train}, X_{test}, y_{train}, y_{test} = \text{train\_test\_split}(X, y, t\_size)$ 
5:   for model in models do
6:     ▷ For classical ML models, extract image embeddings from CLIP/ViT
7:     ▷ For non classical ML models, add an extra classification head to perform the task
8:      $\text{Return Top-5 Accuracy}$ 
9:   end procedure

```

6.2.4 Multi-Modal Input-Output Attack ($\mathcal{A}_{io}$)

Formulation. In this setup, we consider a multi-modal adversary ($\mathcal{A}_{io}$). The auxiliary dataset D_{aux} consists of natural language prompts for a text-to-image model, the cor-

responding generated images, and author labels for each prompt (t, i, α) . $\mathcal{A}_{io}$ aims to learn the mapping $g_\phi : G^T(T) \oplus G^I(I) \rightarrow S$, where the adversary is restricted to learning from the combination of text and image embeddings. This design choice is based on our previous observation that transformer-based image embeddings more effectively capture image representations for this inference task.

Design and Implementation. Similarly to $\mathcal{A}_o$, we leverage CLIP to extract image embeddings. A key property of CLIP is its ability to extract both text and image embeddings, with the crucial advantage that embeddings from different modalities share the same feature space. CLIP’s contrastive learning approach brings images closer to their corresponding text descriptions while pushing dissimilar images further apart in the latent space.

The textual and vision embeddings are combined after flattening the text embeddings. Since both reside in the same feature space, we use simple concatenation—rather than alternative methods like averaging—to effectively enrich the information available to our multi-modal classifiers. We train the same classifiers as in the output attack, denoting the non-data-augmented multi-modal attack as $\mathcal{A}_{io}$ and the data-augmented version as $\mathcal{A}_{io}^+$.

Our $\mathcal{A}_{io}$ pseudocode is presented in Algorithm 5.

6.3 Evaluation Setup

Datasets. We use *DiffusionDB Large* [342], a large-scale text-to-image NLI dataset containing 14 million images generated by *Stable Diffusion* from 1.8 million NLIs (see Table 6.2).

All NLIs were collected from Stable Diffusion Discord channels, and all images were generated using Stable Diffusion Version 1. We preprocess the dataset by removing: i)

Algorithm 5 Input-Output Attack Algorithm ($\mathcal{A}_{io}$)

Input A subset of $U \rightarrow U', I \rightarrow I', T_A \rightarrow T'_A$ and $T_{U_x} \rightarrow T'_{A_x}$
Output Top-5 predictions of U'

```

1: procedure AUTHORSHIP_INFERENCE_INPUT_OUTPUT_ATTACK
2:    $NLIs[] \leftarrow$  All the prompts embeddings of  $T'_U$ 
3:    $\triangleright$  For each author in  $U'$  extract the prompt embeddings of  $T'_U$ 
4:    $Images[] \leftarrow$  All the image embeddings of  $I'$ 
5:    $\triangleright$  For each author in  $U'$  extract the image embeddings of  $I'$ 
6:    $X[] \leftarrow \text{concatenate}(NLIs[], Images[])$ 
7:    $\triangleright$  For each author in  $U'$  concatenate the text and image embeddings
8:    $y[] \leftarrow$  All the authors of  $U'$ 
9:    $X_{train}, X_{test}, y_{train}, y_{test} = \text{train\_test\_split}(X, y, t\_size)$ 
10:  for model in models do
11:     $\triangleright$  For all models, extract text embeddings from CLIP/ViT
12:     $\triangleright$  For classical ML models, extract image embeddings from CLIP/ViT
13:     $\triangleright$  For non classical ML models, add an extra input layer and a classification head to perform
        the task
14:  Return Top - 5 Accucary

```

Table 6.2: Comparison of the size of the DiffusionDB datasets. DiffusionDB Large is a superset of DiffusionDB Small.

Datasets	# of Images	# of Unique NLIs	Image Sizes
DiffusionDB Small	2M	1.5M	Various
DiffusionDB Large	14M	1.8M	Various

exact duplicate NLIs ii) NLIs containing non-English characters, iii) NLIs with at least one *NaN* or *Null* character, and iv) all whitespace. We refer to the resulting dataset as ‘DiffusionDB’. The remaining unique authors and prompts are 10,334 and 1,760,664, respectively. Figure 6.4 shows the distribution of unique authors in “DiffusionDB” by minimum prompt count.

We also use NLIs from ‘DiffusionDB’ to create images with Stable Diffusion (v2.1) and Latent Diffusion to further investigate the behaviour of our spear authorship inferences. To study dementia markers, we use the *ADReSS* dataset [202], which contains speech transcriptions from both a control cohort and a dementia cohort tasked with describing a given image.

Spear Authorship Inference Evaluation Metric. Accuracy is one of the most popular performance metrics in classification problems. More specifically, accuracy is the fraction of total records that are correctly predicted by the model. Top-1 accuracy as-

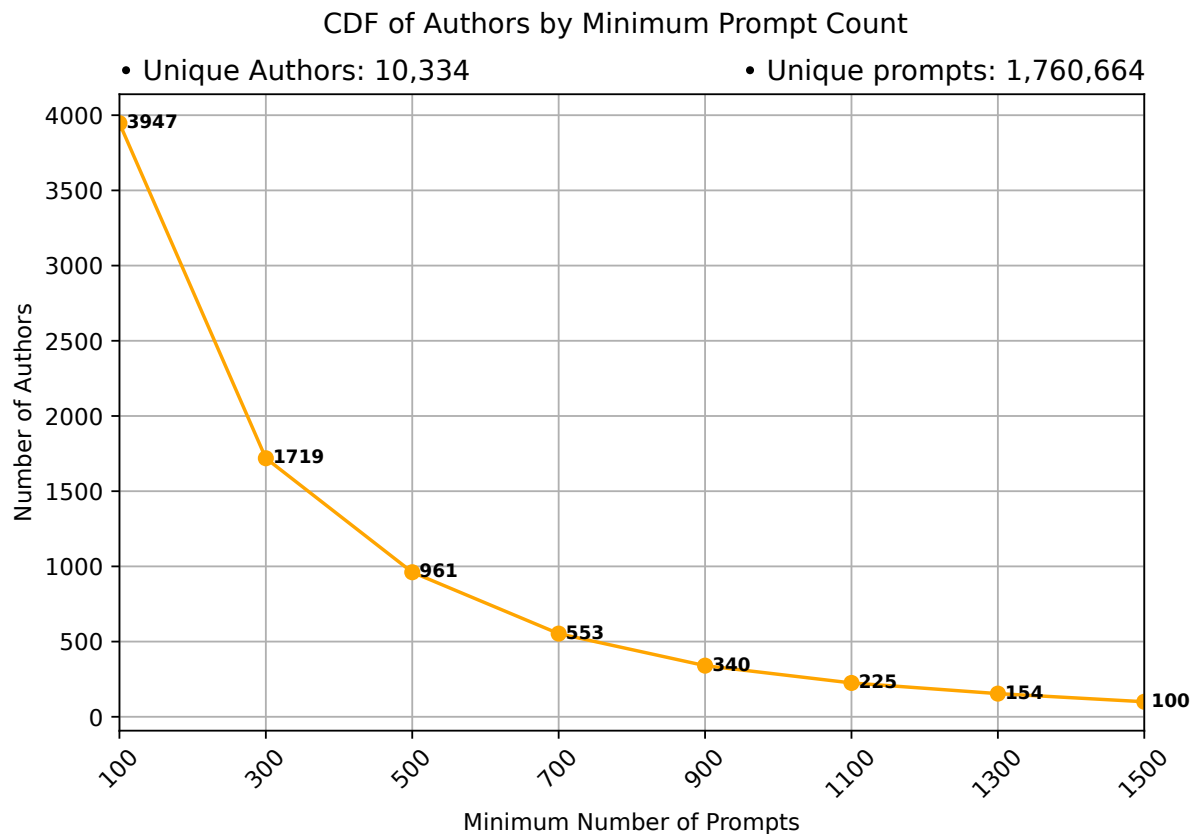


Figure 6.4: CDF of authors by minimum prompt count.

sumes that the model prediction (the one with the highest probability) must be exactly the expected answer. Top-N accuracy, measures how often your predicted class falls in the top N values of your softmax distribution [109]. In several multi-class classification problems like text and image classification, where the classes are difficult to discriminate, Top-N accuracy is a better measure of model performance [302, 291, 31, 273, 153]. Note also that according to GDPR, it is not necessary that the data subject is identified with absolute certainty, and a degree of probability is sometimes sufficient to enable differential treatment of individuals [280]. Therefore, Top-N Accuracy is a good indicative measure of a privacy violation.

To facilitate comparison with prior work [302] below we report Top-5 accuracy. Top-5 provides a more comprehensive evaluation of the models' performance compared to Top-1 accuracy, where only the most probable prediction is considered. Top-5 better reflects real-world use cases: in practical applications of authorship identification, an organization

might be interested in a shortlist of possible authors rather than a single definitive answer. A Top-5 list can be more useful in such contexts, providing a set of plausible candidates. Moreover, Top-5 accuracy reduces the impact of instances where the model makes a single incorrect prediction that could skew the overall assessment of its performance. It provides a more balanced view by acknowledging that the model’s secondary choices are also relevant. However, we also provide a break-down of the Top-(1,2,3,4,5) accuracies for all inference models.

Research Questions. Our evaluation seeks to address the following key research questions:

- **RQ1:** Are T2I spearheaded authorship identification attacks effective?
- **RQ2:** How do the number of available training samples and the size of the target author set affect the attack performance?
- **RQ3:** What properties of NLIs and their generated images can be indicative of authorship?
- **RQ4:** Do these attacks depend on the target T2I model?
- **RQ5:** Can the attacks be adapted to an open-world setting?
- **RQ6:** Do these attacks remain robust against SOTA mitigation strategies?
- **RQ7:** Do T2I models encode other user attributes in their generated images?

6.4 Evaluation

6.4.1 Attack Effectiveness

To answer RQ1, we evaluate the overall effectiveness of our spearheaded attacks in a closed-world setting. For the threat model, we fix the number of authors to $|A| = 100$, a

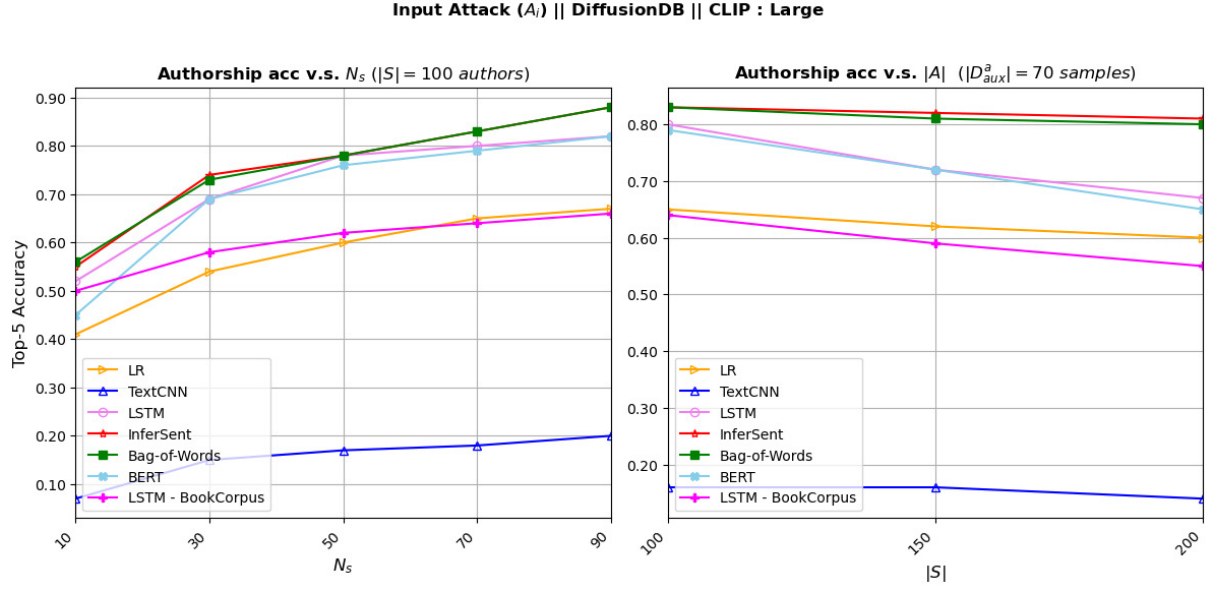
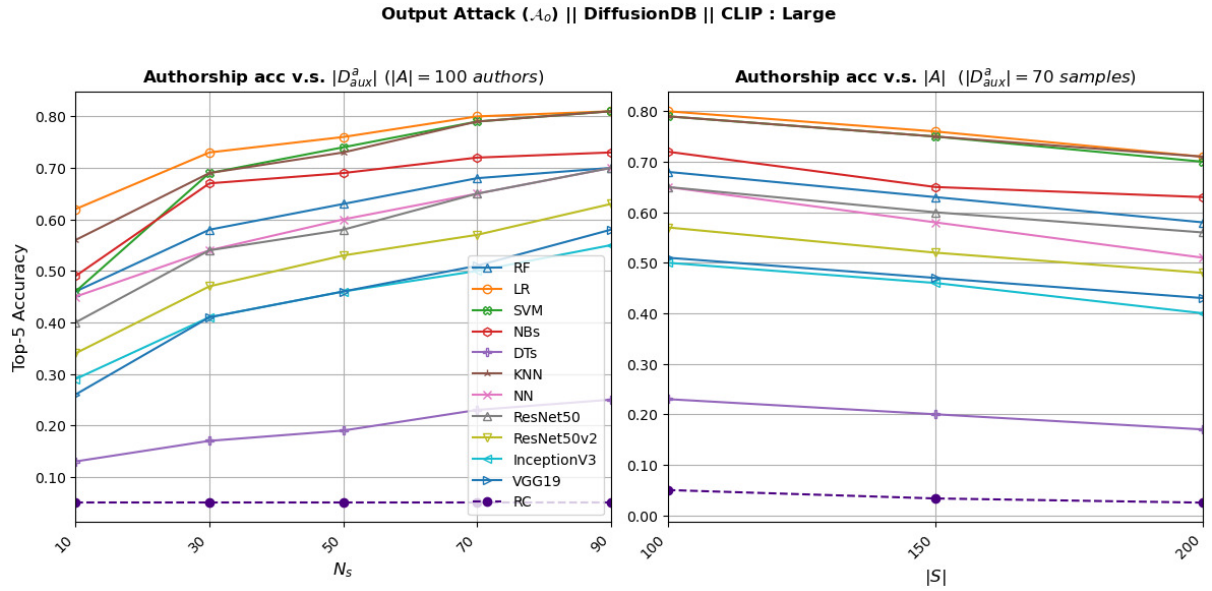
reasonable size for a spearheaded attack. This choice aligns with prior work on authorship identification [302] to facilitate comparison. We then assess the ability of $\mathcal{A}_i$, $\mathcal{A}_o$, and $\mathcal{A}_{io}$ to identify target authors. Specifically, we evaluate 7 $\mathcal{A}_i$ architectures, 11 $\mathcal{A}_o$ architectures, and 12 $\mathcal{A}_{io}$ architectures, as described in Section 6.2.

The models’ performance is compared using a limited number of training samples per author, i.e., $|D_{aux}^\alpha| = 10, 30, 50, 70, 90$. This represents a challenging scenario for the adversary, who, as we will demonstrate, improves only as more labelled samples become available. Authors are selected by first identifying those with at least 100 prompts (3947/10334) in the *DiffusionDB* dataset, and then randomly choosing 100. For the $\mathcal{A}_o$ models, we repeat the experiments twice: once using CLIP embeddings from images and once using ViT embeddings, to evaluate the (in)dependence on the text encoder of the underlying T2I model.

Next, we repeat the experiments with $|D_{aux}^\alpha| = 70$ while varying the number of authors, i.e., $|A| = 100, 150, 200$. This resulted in training and evaluating 49 $\mathcal{A}_i$ models, 154 $\mathcal{A}_o$ models, and 84 $\mathcal{A}_{io}$ models. Although increasing the number of authors beyond 200 is outside the scope of our primary attack, we conduct some preliminary experiments and provide insights into the adversary’s scalability beyond this point.

Input Attack ($\mathcal{A}_i$) Effectiveness. Our InferSent, and BERT embedding-based classifiers (referred to as “InferSent” and “Bert”), and our Bag-of-words based classifier (referred to as “BoW”) are compared against one of the best performing model from Song and Raghunathan [302] (LSTM_BookCorpus). We also compare against an improved version trained on our dataset (LSTM). We also compare against a Logistic Regression (LR) multi-class classifier trained with text embeddings extracted with CLIP [241] a vision transformer model. Lastly we compare against TextCNN [161] which has shown success in other authorship attribution works [272, 293].

Our results are summarized in Figure 6.5a. We found that all our classifiers (InferSent, BERT, LSTM_DiffusionDB and BoW) perform better than the other models, signifi-

(a) Input Attack ($\mathcal{A}_i$).(b) Output Attack ($\mathcal{A}_o$), using CLIP embeddings.

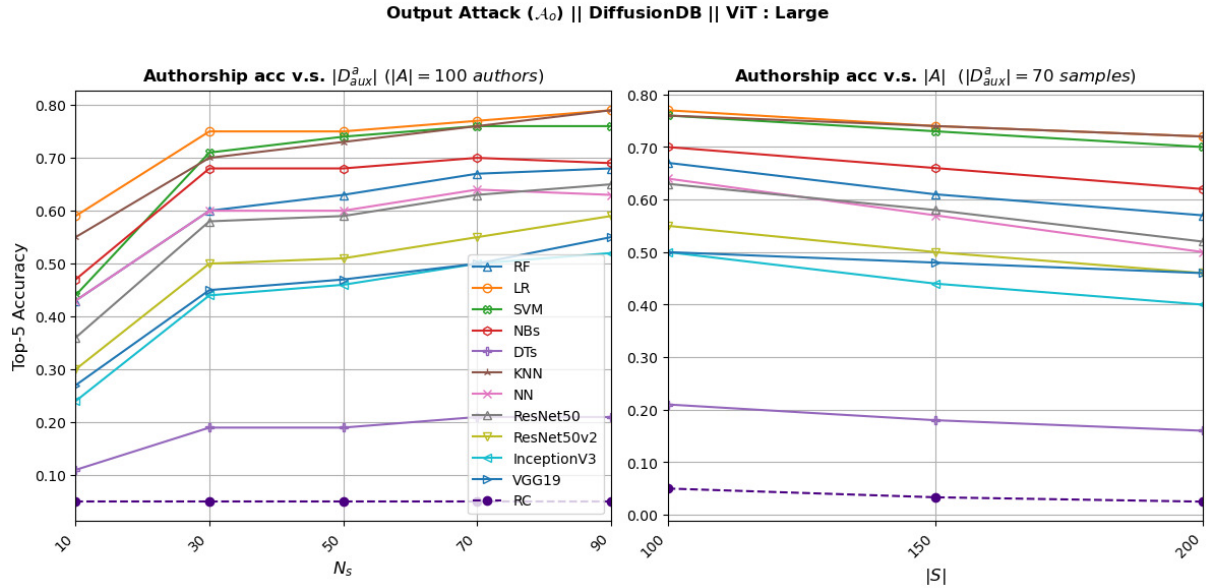
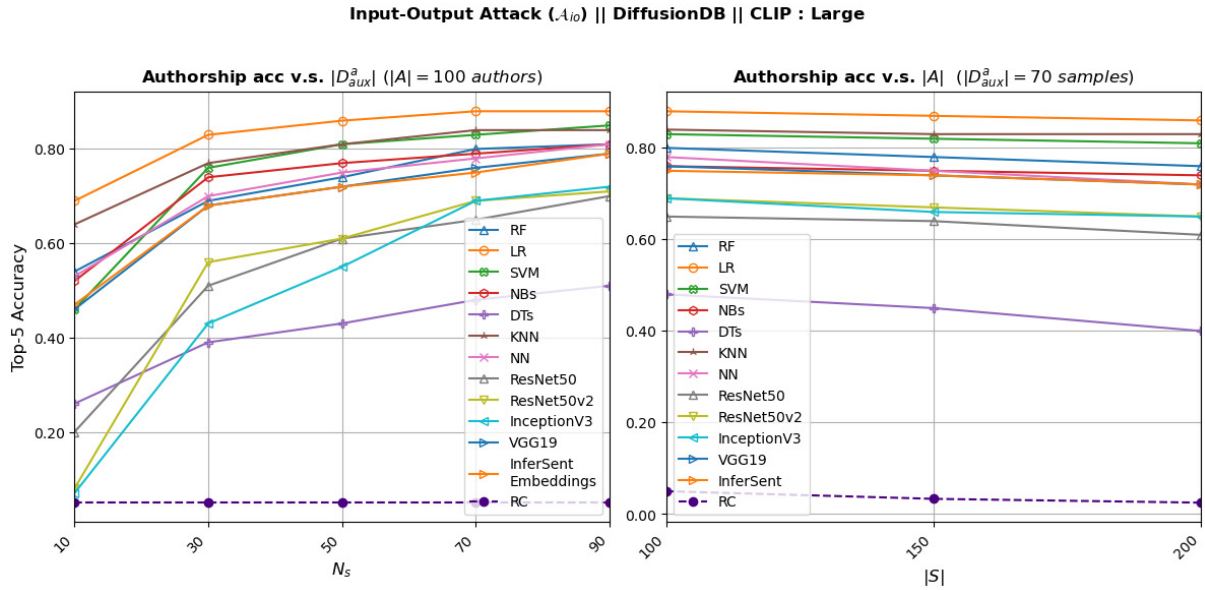
(c) Output Attack ($\mathcal{A}_o$), using ViT embeddings.(d) Multi-Modal Input-Output Attack ($\mathcal{A}_{io}$), using CLIP embeddings.

Figure 6.5: Comparison of all adversary models on DiffusionDB.

cantly surpassing the performance of our TextCNN. LR and LSTM_BookCorpus baselines. LSTM trained on our dataset performed better compared to the baselines and close but still worse compared to our attack models. InferSent exhibits the best performance with 0.77 top-5 accuracy when leveraging only 70 labelled NLI for training per author with the performance gradually improving as we add more training samples. It is worth noting here that in contrast with prior work in authorship identification from text [302], we find that even with limited training samples, approaches such as BoW which are not dependent on pre-trained unsupervised embeddings also generalize and perform well in our task. This has to do with the nature of the text in NLIs. NLIs have more constrained stylistic properties, for example all NLIs aim to describe an image, authors use NLIs to generate a specific set of images with a specific style, while NLIs on average have shorter sentence length compared to sentences from books. A more detailed analysis of this can be found in our project’s website. Note also that the LR model trained on multi-modal CLIP embeddings performed well but worse than our models. This signifies that while not the best $\mathcal{A}_i$ model, CLIP text embeddings can still be useful in multi-modal classification as we show in the next sections.

Increasing the number of authors ($|A| = 200$) but with a small number of training samples per author ($|D_{aux}^\alpha| = 70$), the InferSent and BoW models remain robust while the performance of the rest of the models starts degrading.

Output Attack ($\mathcal{A}_o$) Effectiveness.

As explained in Section 6.2, our adversarial models use CLIP-large [241] and ViT-large [352] to extract image embeddings which are then fed into some of the most popular classical classification algorithms : Support Machine Vector (SVM) [59], Logistic Regression (LR) [61], Random Forest (RF) [130], Naive Bayes (NBs) [209], Decision Tree (DTs) [353], Neural Network (NN) [116], and K-Nearest Neighbor (KNN) [228]. All classifiers are configured using the default parameters of the scikit-learn library [248].

We compare these with traditional CNN-based models for image feature embedding ex-

traction. In particular we evaluate a Residual Network-50 (ResNet50) [125] a 50-layer convolutional neural network pre-trained on ImageNet-1k [274] for image classification tasks. Instead of passing the image embeddings from CLIP/ViT as input to a classifier, as we do on our attack models, we first use the ResNet50 baseline to extract the image features from the input images. For better generalization, we also apply image perturbations using *ImageDataGenerator* [158] from Keras. Then, since ResNet50 has not been trained on authorship identification task, we add a custom classification head and fine-tune the model. To avoid overfitting or underfitting the model by setting a very large or small number of epochs, respectively, we use *early stopping* considering the validation loss as performance measure to terminate the training. We set the total number of training epochs to 100, the patience to 3 epochs and the validation split to 0.2. We apply the same procedure to other SOTA pre-trained CNN models, including ResNet50V2 [126], InceptionV3 [311] and VGG19 [155]. Lastly, since authorship attribution from T2I generated images is a novel task we also denote with “*RC*” the random probability (Random Choice) of an author being in the Top-5 predicted authors.

Our results are summarized in Figures 6.5b and 6.5c. Note that the results presented in this Section are from attack models without data augmentation. Nonetheless, we observe that it is possible to identify NLI authors from T2I generated images with high Top-5 accuracy. We also observe that training simple classifiers from both CLIP and the more general ViT embeddings yields better results compared to classifying directly on more traditional image feature embeddings extracted with high performing CNNs on image classification. Logistic Regression from CLIP and ViT embeddings performed systematically better. For example, Logistic Regression with CLIP embeddings on 70 training samples per author, outperforms all the models with 0.79 Top-5 accuracy (see Figure 6.5b). Using ViT embeddings, Logistic Regression beats the rest models with 0.77 Top-5 accuracy on 70 training samples (see Figure 6.5c).

As we increase the number of authors ($|A| = 200$) but with a small number of training samples ($|D_{aux}| = 70$), Logistic Regression and SVM from CLIP and ViT embeddings

remain the highest performing attacks but with a drop in performance.

Input-Output Multi-Modal Attack($\mathcal{A}_{io}$) Effectiveness. We follow a similar approach for evaluating this attack scenario as with the output adversary ($\mathcal{A}_o$). As explained in Section 6.2 instead of only learning from the image embeddings, our models concatenate these with text embeddings from the same feature space. For comparison, we use the baseline classification models from $\mathcal{A}_o$. However, these are modified for multi-modal learning. In particular, we use the models in $\mathcal{A}_o$ to extract the image feature embeddings which are then concatenated with text embeddings extracted with CLIP after flattening them to match the image embedding dimensions.

As before we annotate the results with a *RC* adversary since this is a novel adversarial task and we want to provide a naive baseline to better appreciate the improvement that more sophisticated adversarial models can achieve.

Our results are summarized in Figure 6.5d. Similarly with the $\mathcal{A}_o$ case, leveraging CLIP embeddings (both for NLI and images) outperforms the other models. For example, Logistic Regression with 70 training samples per author, surpasses the other models with 0.88 Top-5 accuracy. Comparing with $\mathcal{A}_o$ scenario we see a 11% increase on performance under the same experiment parameters. With 200 authors, the models remain robust—the highest Top-5 accuracy slightly drops to 0.85. In general the multi-modal $\mathcal{A}_{io}$ adversary performs better than both the single modality image-only $\mathcal{A}_o$ and text-only $\mathcal{A}_i$ adversaries. The classifiers also exhibit better robustness to an increase in the number of target authors.

Next, Table 6.3 presents Top-1 to Top-5 accuracy results of the inference models across three setups $|A| = 100, 150, 200$ with a fixed training sample size of 70. Accuracy generally improves from Top-1 to Top-5 across all inference models ($\mathcal{A}_i$, $\mathcal{A}_o$ and $\mathcal{A}_{io}$), indicating model performance consistency as more candidate labels are allowed. For example, in the $|A| = 200$ case, $\mathcal{A}_{io}$ achieves 75.30% Top-1 accuracy, reaching 85.84% at Top-5, highlighting its robustness across ranks. Also, in the case of $|A| = 150$, the Top-1 accuracy

for $\mathcal{A}_i$ is 68.91%, while $\mathcal{A}_o$ drops to 60.05%. This shows a clear performance gap, indicating that the input space alone provides more precise author identification than using only the output space. In contrast, our multi-modal inference model $\mathcal{A}_{io}$ performs significantly better, achieving 77.74% Top-1 accuracy in the same scenario. This demonstrates the benefit of leveraging both input and output space.

Table 6.3: Top 1-to-5 accuracy scores on different experiment setups.

Experiment Setup : $\mathcal{D}_{aux} = 70$ with $\mathcal{A} = 100$					
	Top 1	Top 2	Top 3	Top 4	Top 5
$\mathcal{A}_i$	68.91%	75.14%	77.84%	79.91%	81.61%
$\mathcal{A}_o$	60.05%	68.54%	73.24%	76.77%	79.44%
$\mathcal{A}_{io}$	77.74%	82.47%	85.07%	86.77%	88.04%
Experiment Setup : $\mathcal{D}_{aux} = 70$ with $\mathcal{A} = 150$					
$\mathcal{A}_i$	68.56%	74.05%	77.05%	78.72%	80.25%
$\mathcal{A}_o$	54.99%	64.16%	68.78%	71.87%	74.58%
$\mathcal{A}_{io}$	76.58%	81.74%	84.18%	85.74%	87.14%
Experiment Setup : $\mathcal{D}_{aux} = 70$ with $\mathcal{A} = 200$					
$\mathcal{A}_i$	68.62%	73.29%	75.75%	77.42%	79.07%
$\mathcal{A}_o$	52.54%	61.01%	65.86%	69.10%	71.90%
$\mathcal{A}_{io}$	75.30%	80.35%	83.23%	84.97%	85.84%

NSFW Content. The inference models could be potentially used for attribution of not safe for work (NSFW) content created with T2I models. Here we evaluate the attribution capabilities of each of the $\mathcal{A}_i$, $\mathcal{A}_o$ and $\mathcal{A}_{io}$ models. We classify an image as NSFW if it attains a safety score of ≥ 0.7 . Given the limited quantity of *unsafe* content within *DiffusionDB Large*, setting this threshold allows us to identify 115 authors, each associated with a minimum of 100 *unsafe* images. After manual observation, we found that a significant number of images include nude content. Table 6.5 shows the best performing classifiers. We note that when $|\mathcal{A}| = 100$, Logistic Regression excels with 0.74 and 0.86 Top-5 accuracy under $\mathcal{A}_o$ and $\mathcal{A}_{io}$, respectively. These results demonstrate that the

inference models derived with our method can be an important tool to facilitate NSFW content attribution.

Caption models for inferring authorship from T2I-generated images. To explore whether simply applying SOTA caption models on T2I images to recover the original NLI and potentially authorship information we performed an experiment with three state-of-the-art caption models, namely BLIP[189], BLIP2 [188] and LLaVA [193]. We use these models to caption text from 10,000 images randomly sampled from DiffusionDB dataset. We then compare the semantics of the original NLIs and the captions using the *all-MPNet-Base-v2* model. The results are summarize on Table 6.4. BLIP-2 offers the best similarity score with 0.41. However, the semantic results are quite low for all caption models, which indicates that directly inferring authorship from captions is challenging.

In seeking to explain these low similarity scores, we identify three reinforcing factors. First, modern caption models—BLIP, BLIP-2, LLaVA—are pretrained on a mix of human-written and machine-generated descriptions of *natural* photographs [189] [189] [193]; when faced with synthetic images, their outputs degrade because of a pronounced domain shift that has been noted in domain-generalisation studies [378] [249]. Second, the prompt-inversion literature [65] shows that even architectures adapted specifically for reverse-engineering Stable Diffusion images struggle to predict more than a coarse semantic embedding of the prompt, underscoring the ill-posed nature of full prompt recovery. Third, recent work on caption-guided augmentation reports that, without fine-tuning on synthetic data, BLIP-2 tends to produce short, generic sentences and suppress the very stylistic or proprietary tokens that would be most informative for authorship inference [270]. Taken together, these factors show why standard captioning models still fall short of connecting a generated image back to its original prompt.

Table 6.4: Embeddings similarity between the original NLI and the generated image’s caption produced with BLIP, BLIP-2 and LLaVA.

	Pre-trained Image Captioning Models		
	BLIP	BLIP-2	LLaVa
Mean Similarity	0.35	0.41	0.35

Table 6.5: The Top-5 accuracy of the best performing (in parenthesis) Authorship Inference models using NSFW images.

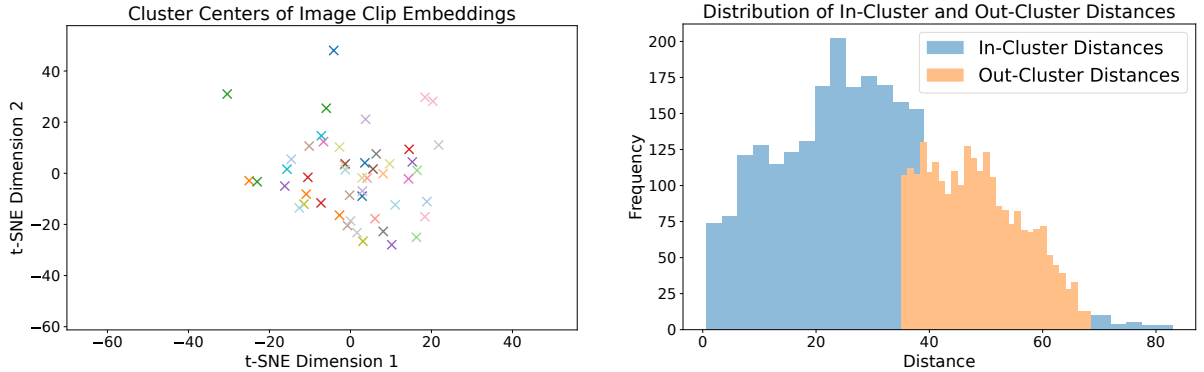
Experiment Setup : $\mathcal{D}_{aux} = 10...90$ with $\mathcal{A} = 100$									
	Training Samples								
	10	20	30	40	50	60	70	80	90
$\mathcal{A}_i$ (InferSent)	0.61	0.72	0.76	0.78	0.80	0.81	0.83	0.84	0.85
$\mathcal{A}_o$ (LR)	0.61	0.67	0.72	0.73	0.73	0.74	0.74	0.75	0.77
$\mathcal{A}_{io}$ (LR)	0.69	0.77	0.81	0.83	0.84	0.85	0.86	0.86	0.87

6.4.2 Leakage Analysis

In this section, we try to delve deeper into our understanding on why it is possible to infer authorship in T2I and more specifically, we want to explore why leakage is possible through generated images.

The importance of image embeddings for author separability. To analyse the output space of T2I models for authorship separability, we focus on CLIP image embeddings of images from DiffusionDB. Specifically, we randomly sample 50 authors from our training set of 100 authors, and for each author we randomly sample 50 NLIs. The choice of 50 instead of 100 which was used in the attacks is merely because of computation time and because it is easier to visualize. Then we extract the CLIP embeddings for each generated image. CLIP embeddings are multi-dimensional and also hard to visualize. To address this we use t-SNE [329] (t-distributed Stochastic Neighbor Embedding) which is a widely used unsupervised dimensionality reduction technique, to reduce a CLIP image embedding dimensions to two. Then we calculate per-author centroids of

CLIP embeddings by averaging across the two dimensions and illustrate the results in Figure 6.6a. Lastly, we calculate the in-cluster and out-cluster Euclidean distances. Figure 6.6b shows the per-author cluster centroids of CLIP image embeddings after t-SNE dimensionality reduction, along with the in-cluster and out-cluster Euclidean distance distributions. When in-cluster distances are small, it indicates higher similarity among CLIP embeddings, while out-cluster distances tend to be larger, reflecting greater dissimilarity between clusters. These results show that separability is challenging on the output of T2I models which is expected due to the lossy image generation process which is inevitable due to the imperfect alignment between instruction and image. Nonetheless it is still possible with cluster centroids not overlapping. Also the P_{25} , P_{50} , P_{75} , P_{90} for in-cluster distances and out-cluster distances are 17.6, 28.0, 41.6, 53.4 and 43.2, 50.0, 57.4, 63.1 respectively with the minimum out-cluster distance is 36.7 and between the median and 75% of in-cluster distances are still below that threshold. Taken together, these distance statistics quantify *how much* stylistic leakage exists; the next paragraph examines *why* those numbers could be inflated by topical bias—that is, when authors simply generate different subjects rather than different styles.



(a) t-SNE Cluster Centers of per-author CLIP image embeddings. (b) In-Cluster and Out-Cluster Distances of per-author CLIP image embeddings.

Figure 6.6: Analysis on 50 authors with 50 NLIs each (a,b) and 50 images generated from those NLIs using DiffusionDB dataset.

Illustrative example of prompt-image-author linkage. Figure 6.7 shows the prompt “pig in a red sweater flying in a cyber-punk city” together with the generated image selected from our 50×50 sample. The image-only classifier A_o lists author α_{42} as its

Top-1 prediction (within the Top-5 shortlist), which is the correct author. The example illustrates how the author’s characteristic colour palette (desaturated cyan cityscape plus bold red garment) and preference for symmetrical framing are captured in the CLIP embedding, enabling accurate attribution from a single output image.

Threats to validity: topical

bias and user interests. In particular, some of the inter-author separation we see in Figure 6.6 could be driven by subject-matter preferences (topical bias) rather than stylistic fingerprints. Specifically, clusters in Figure 6.6 may partly reflect *what* users like to prompt (topics) rather than purely *how* they write or the stylistic fingerprint of their images. If two authors repeatedly request dragons versus sunsets, separability could arise trivially from



Figure 6.7: Author’s Prompt: “pig in a red sweater flying in a cyber-punk city”. A_o : Top-1 = α_{42} (in Top-5).

object categories. We therefore replicated the analysis on a *topically homogeneous* slice: the NSFW subset where almost every image depicts human nudity (see Table 6.5 in Section 6.4.1) and topics are narrowly constrained. Even there, when $|A| = 100$ the A_o and A_{i_o} achieve 0.74 and 0.86 Top-5 accuracy with 70 training samples, respectively. This indicates that, although user interests do contribute signal, author-specific stylistic leakage in the *image space* remains substantial. Future work should still examine longitudinal drift: as users’ interests evolve, the clustering boundaries (median in-cluster 28.0 vs. minimum out-cluster 36.7) may contract, lowering attack success and requiring periodic retraining of A_o .

Discussion on content dependence. Other works on authorship inference showed that authors’ stylometric features might be correlated with the text topic [224] and more generally the content of the written text might be biasing classification. Our experiments show that it is still possible to perform classification on authorship and dementia inference even on similar content. More specifically, our experiments with NSFW images conduct authorship attribution on only NSFW content, while our dementia inference model (Section 5.5.1) is trained on descriptions of the same image. In the case of authorship inference it is still possible that different authors might focus on different topics. Therefore, the inference models might need to be retrained periodically as the target users’ interests shift. We do not expect this to happen very often and existing models dependent on user interests are successfully deployed in practice (e.g., personalized advertising).

6.4.3 Model Diversity

To answer RQ4 on whether our attacks depend on the T2I model we design a separate experiment. This experiment is performed on the output attacks because the effect of the T2I model (generated images) is better isolated if we focus only on the output. By definition input attacks are independent of the T2I model, and input-output attacks can only be better than output attacks. In this experiment we want to analyse the performance of our attack models when trained on images of a T2I model and perform inference on images of the same model (classifier-dependent attacks), and when trained on images of a T2I model and perform inference on images of another model (classifier independent attacks). We evaluate models trained and tested on images from both Stable Diffusion V2.1 and Latent Diffusion, and to show the benefits of data augmentation we train the models without and with data augmentation. This totals 8 different settings for which we evaluate all our attack models from Section 6.4.1. Across all experiments, we leverage CLIP embeddings for the images, and we consistently apply a 70 : 30 training/test split.

To generate the datasets for the non-augmented attacks, we use the same set of 100

Table 6.6: Top-5 Accuracy of Best Performing Output Attack ($\mathcal{A}_o$) models without and with data augmentation in model-dependent (*) and model-independent inference settings.

Training Images		Test Images on T2I Models	
T2I Model	Data Augmentation	Stable Diffusion v2.1	Latent Diffusion
Stable Diffusion v2.1	NO	0.559* (LR)	0.541 (LR)
	YES	0.695* (KNN)	0.697 (KNN)
Latent Diffusion	NO	0.553 (LR)	0.553* (LR)
	YES	0.692 (KNN)	0.694* (KNN)

authors with 100 NLIs each from Section 6.4.1 to generate 10,000 new images using the open-source and free Latent Diffusion [269] and Stable Diffusion V2.1 [268] models. To generate the data augmented datasets we repeat the above but instead of generating one image for each NLI we generate 5 image variants per NLI, resulting in 50,000 image samples in total. Table 6.6 summarizes all our results where we only report the best performing output model for clarity.

We first observe that non data-augmented output attacks ($\mathcal{A}_o$) do not achieve the same performance as the non data-augmented output attacks on DiffusionDB (see Section 6.4.1). We hypothesize that this is because the authorship content might transfer better in the DiffusionDB (Stable Diffusion v1 model). To examine this hypothesis we use the CLIP score to evaluate the similarity between the NLI and the output images (non-augmented datasets) from DiffusionDB, Stable Diffusion v2.1, and Latent Diffusion. To calculate the CLIP scores we use the multi-modal CLIP-Large [241] to extract text and image embeddings from NLIs output images respectively. Since the embeddings are in the the same vector space we can then calculate their cosine similarity. The CLIP score ranges between [0..100] and higher scores indicate higher similarity. Figure 6.8 shows the cumulative distribution function of the CLIP scores for the three models. Given prior work [122], a CLIP score around to 0.26 indicates that a generated image has good alignment to its input prompt. Clearly, DiffusionDB has significantly more images (around 80%) with score ≥ 0.26 than Stable Diffusion v2.1 (around 55%) and Latent Diffusion (around 20%).

This might explain why the non-augmented attacks are more successful on DiffusionDB images. However, as shown on Table 6.6 our data-augmented attacks ($\mathcal{A}_o^+$) achieve high Top-5 accuracy in model-dependent and model-independent settings and across training sets.

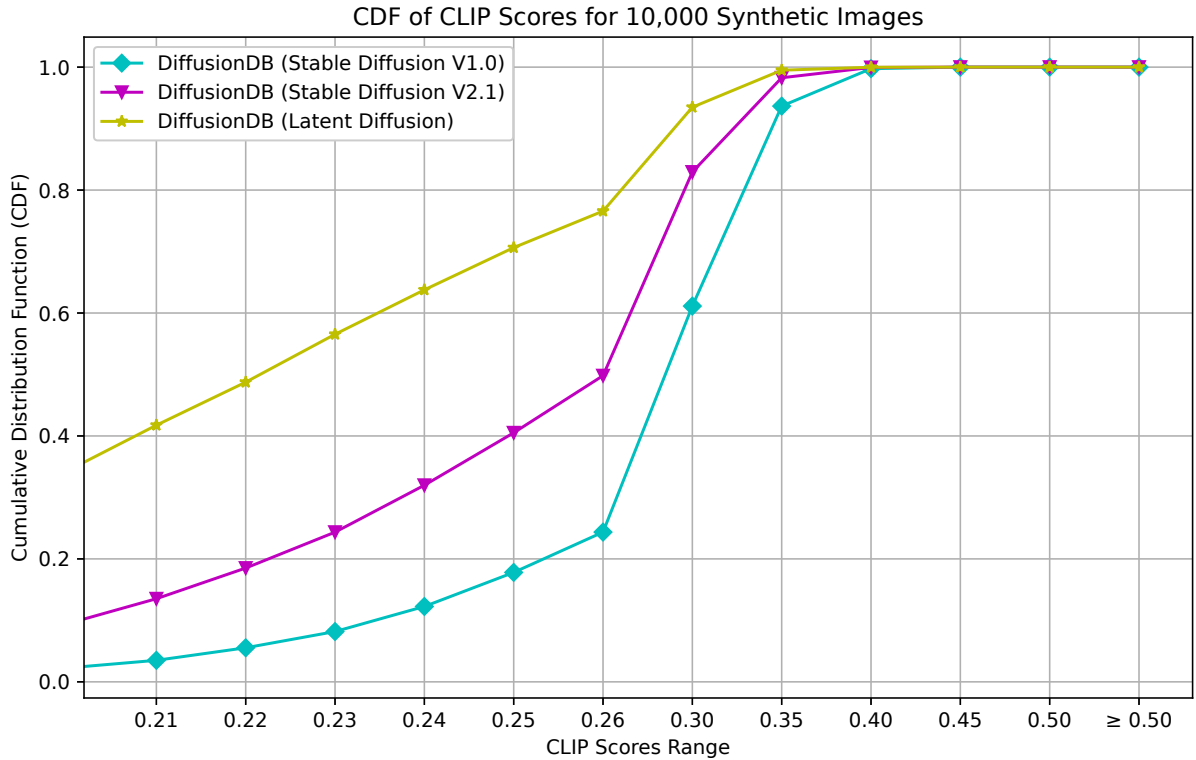


Figure 6.8: CDF of CLIP scores for 10K images per model.

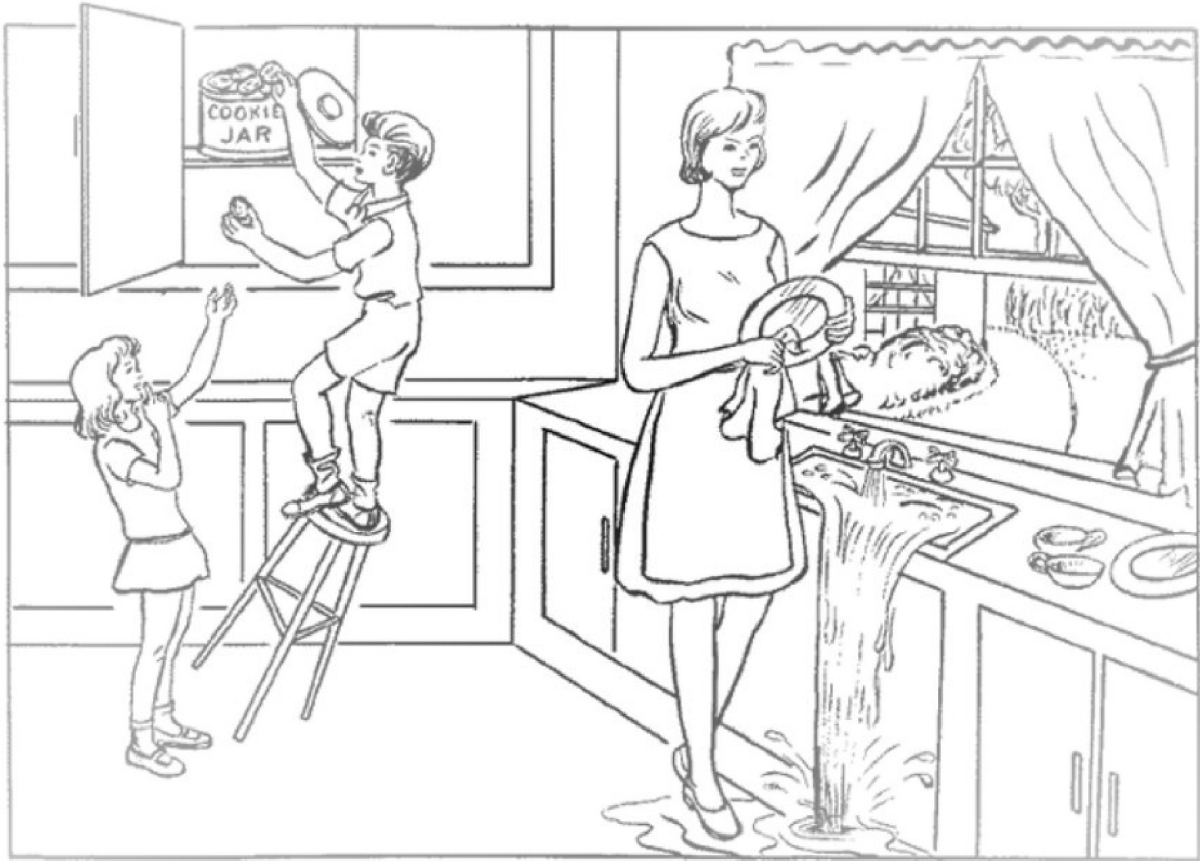
6.4.4 Leakage of other attributes

We have demonstrated that authorship information can be leaked through both the input (natural language instructions) and the output (generated images) of text-to-image models. However, other attributes may be at risk as well (RQ7). Inspired by the fact that pathogenic markers are identifiable via text [25, 388] here we study whether they survive the diffusion process and appear in the generated images. This should be possible, if such language markers are embedded in the joint text-image latent space used to guide the denoising process. Then if the denoising process is accurate, then it is possible that the markers will affect the respective image. To study this effect, we leverage an existing

dataset with manual transcriptions of image descriptions from a control group and a group suffering from Alzheimer’s disease. Alzheimer’s disease is a type of dementia neurocognitive condition which affects (very conservatively) tens of millions of vulnerable users worldwide with its number of cases expected to triple by 2050 to over 150 million [319], and its leakage from technology usage has not been studied before. Alzheimer’s disease is extremely sensitive with patients vulnerable to manifestations of restlessness and agitation which require management through costly antipsychotic drugs. Leaking one’s condition can be manipulated for targeted advertising, spear phishing attacks, bias in the workplace and others and thus constitutes a major privacy concern. Note that the descriptions are not NLI and do not represent what the participants would write to describe an image. They are however representative of a scenario, where a voice interface is used to provide instructions to an T2I model. Such interfaces are already widely used [92, 246].

Dataset. The ADReSS [202] dataset, is a subset of the DementiaBank dataset [28] that is pre-processed and balanced in terms of age and gender. Participants are diagnosed with various stages of dementia. During the experiment, 156 participants are provided with the same “Cookie Theft Picture” shown in Figure 6.9 and are asked to describe it orally. The descriptions are manually transcribed eliminating the effect of Automatic Speech Recognition (ASR) errors. The ADReSS dataset contains healthy-control (CC) and dementia-labelled (AD) descriptions, with 54 train and 24 test samples for each class, for a total of 156. We further preprocess the dataset to remove irrelevant sentences. For example, after manual analysis, we observed that the descriptions might contain clarification questions irrelevant to the task of describing an image. Two such examples are shown in Figure 6.9a and Figure 6.9b. We manually remove all such instances from the transcriptions to ensure the descriptions are more valid and focused.

Experimental Setup. To test whether dementia-related language markers are encoded in joint text-image representations used in text-guided diffusion models, we use the same black-box approach used to study spear authorship leakage. In particular, we first verify that dementia can be inferred from text and reproduced the text-only classifier from

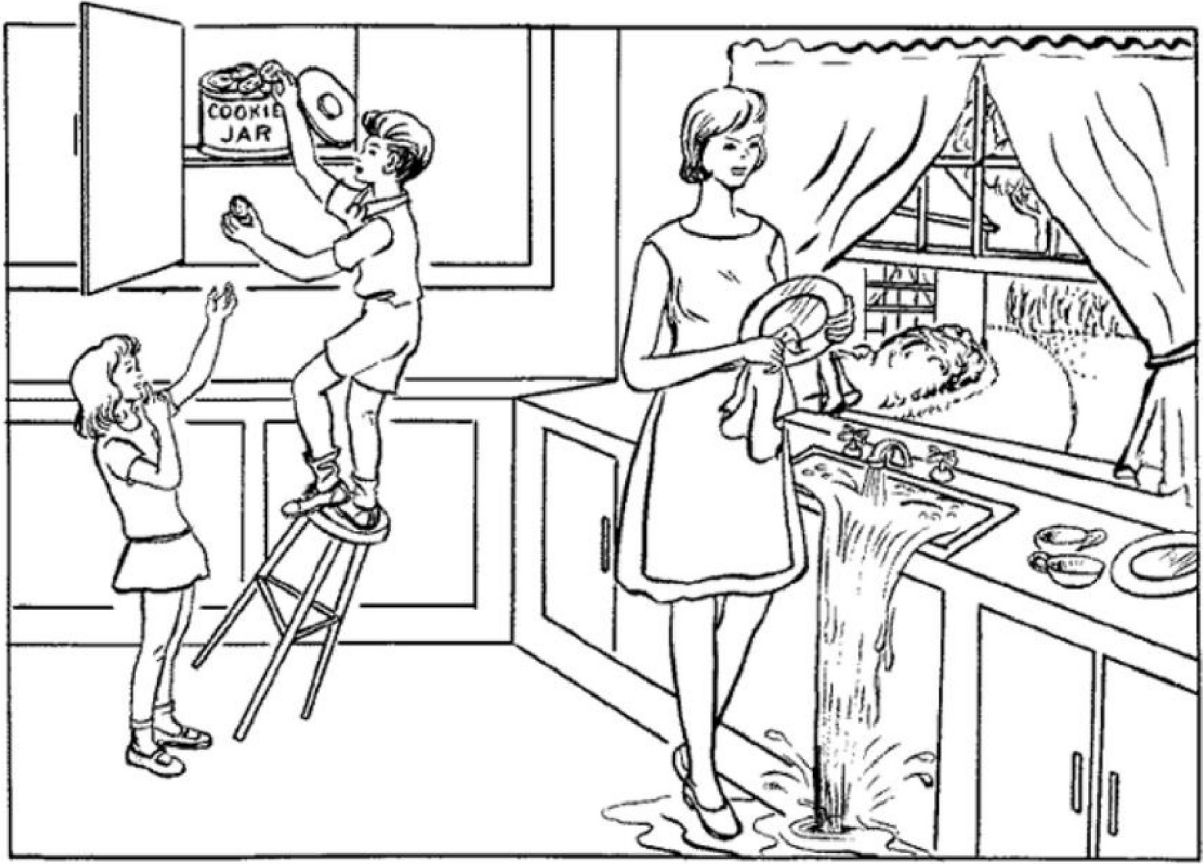


okay.there's a little boy and he's getting he's standing on a stool that's upsetting.and he's getting a cookie and he's sharing a cookie with the little girl who is shushing her mother with her finger and reaching for the cookie at the same time.on the k...what are what are the instructions?just give as many details like?action.or d or.oh okay.the cupboard door is open.the stool is crooked.the water is splashing on the floor.the m uh mother is um I'm assuming it's a mother is stepping in it.she's drying her dishes.the uh sink is overflowing.the cups and the saucers are there.the window is open and you see the lawn outside.and it must be summertime.they all have on shor short sleeves.and uh she's wearing an apron.okay.

(a) First example of a transcription of an *AD* patient.

Balagopalan et al. [25] which achieves SOTA results in dementia classification from transcribed speech.

This approach though uses BERT embeddings. To study whether the dementia features are encoded in text-image embeddings which are used in text-guided image generation with diffusion, we train A_i^d dementia inference models on the CLIP embeddings extracted from the ADReSS cleaned image descriptions. We train the models for 100 epochs using



mhm.well the water's running over on the floor.uh the chair is tilting.the boy is into the cookie jar.and his sister is reaching for a cookie.the mother's drying dishes.um **do you want action or just want anything I see?** okay.mhm.

(b) Second example of a transcription of an *AD* patient.

Figure 6.9: Two examples of transcriptions of *AD* patients describing the Cookie Theft picture shown.

the Binary Cross Entropy Loss. The A_i^d model will show whether joint representations encode dementia language markers. To further show whether the final generated images guided by such representations can leak dementia, we need to further study the success of dementia classifiers trained on the generated images (A_o^d). The ADReSS dataset only provides the image description (text). We use the descriptions to generate images with Stable Diffusion v2.1. Then we extract CLIP embeddings from the images and train all model architectures used in A_o spear authorship inferences. We use a 70 : 30 training/test split (20% of the training set is used for validation to avoid overfitting to the test set)

because of the small size of the *ADReSS* dataset. Lastly, for completeness we also retrain all the A_{io} models on the concatenated text and image embeddings using the same training settings as above to derive the A_{io}^d .

Table 6.7: The accuracy of the best performing (in parenthesis) Dementia Inference models using images from Stable Diffusion v2.1 and Latent Diffusion.

	$\mathcal{A}_i^d$	$\mathcal{A}_o^d$	$\mathcal{A}_{io}^d$
Stable Diffusion v2.1	0.833 (NN)	0.75 (SVM)	0.797 (RF)
Latent Diffusion		0.729 (NN)	0.771 (NN)

Discussion. The highest performing models are shown on Table 6.7. Our findings demonstrate that pathological markers in textual image descriptions are encoded in joint text-image representations as shown at the high accuracy (83.3%) of the $\mathcal{A}_i^d$ model in differentiating between dementia and control descriptions from such representations. Moreover, our $\mathcal{A}_o^d$ results (75% accuracy) show that the images generated with diffusion-based T2I models guided by such representations can themselves contain indications of the condition which reveals a previously unidentified privacy risk with the usage of such models by vulnerable populations. Finally, the $\mathcal{A}_{io}^d$ accuracy is close to 80% which improves compared to $\mathcal{A}_o^d$ but drops compared to $\mathcal{A}_i^d$. We further verified that by removing pathological language markers such as discourse tokens (e.g., “um”, “well” etc.) which were shown before to be indicative of dementia in speech and language, reduces classification accuracy in all settings with drastic decreases of more than 10% in the $\mathcal{A}_o^d$ and $\mathcal{A}_{io}^d$ settings.

6.4.5 Code Availability

An implementation of the authorship inference attacks described in this chapter are available in an APSS-Lab repository under the MIT open-source licence available at GitHub [174]. See the ‘README.md’ file for instructions on how to see the code used and reproduce the results in this chapter.

6.5 Robustness Against Obfuscation

Our evaluation (Section 6.4) shows that adversaries can construct input (text), output (image) and multi-modal input-output (text-image) attacks to infer authorship from NLI and their T2I images. Here we explore the robustness of our attacks against SOTA mitigation mechanisms. These results are summarized in Section 6.5.2.

6.5.1 Setup and Mitigation Approaches

Authorship Obfuscators. Authorship inference have been an issue traditionally studied in the written language domain. In this study we showed that leakage can happen from both the input (text) and the output (image) domain of text-to-image models. However, it is reasonable to assume that if obfuscation is successful on the input domain, then the generated images from the obfuscated instructions will also be harder to distinguish between authors.

Prior works have proposed text obfuscation strategies that can be generally classified into either learning anonymous textual representations [347] or developing text transformation mechanisms that can obfuscate authorship and simultaneously retain the semantics or meaning of the original input [340]. The latter category is more appropriate in our case since it is easier to integrate in a T2I scenario.

Given the above, to evaluate the most promising defenses, we chose a SOTA text paraphrasing model (PEGASUS [380]) as our baseline obfuscator. Such models tend to achieve good diversity in text generation while achieving high preservation of the original semantics of the text. Nonetheless, they are not designed for obfuscation tasks. To mitigate this, we leverage the observations by Mattern et al. [216] who show that by varying the *temperature* in the word sampling stage of language models can be used to inject noise, offering a privacy-utility control knob. In our implementation, we apply the pre-trained

paraphrase model PEGASUS, to generate paraphrases of maximum length $n = 60$ while using different temperatures $T = 1, 1.5, 2, 3, 5, 10$ and 15 . Mattern et al. [216] show that sampling a token from a soft-max distribution with temperature T realises the Exponential Mechanism with privacy budget $\epsilon = \frac{2\Delta_q}{T}$, where Δ_q is the ℓ_∞ -sensitivity of the quality function (*Equation 5* in their paper). Because they rescale all logits to the interval $[0, 1]$, one obtains $\Delta_q \leq 1$ and therefore the conservative bound $\epsilon \leq 2/T$. The rule-of-thumb $\epsilon \approx 1/T$ emerges only under an additional, application-specific assumption that $\Delta_q = \frac{1}{2}$ —for example, if the utility range is further compressed or a binary indicator score is used—an assumption *not* required by their proof. Accordingly, we treat T as an adjustable privacy-utility knob and report the resulting trade-offs empirically in Section 6.5.2.

Moreover, given the stupendous success large language models (LLMs) exhibit in various tasks, we consider some of the most performing open source large language models (LLMs) to construct paraphrases of the original NLI [300]. In our work, we employ and evaluate the Llama3 and Mistral models. We chose these because they are free and open-source models. Both of them have shown robust performance on downstream tasks [19, 382]. Specifically, a single request was made for each unique NLI, which consisted of a fixed instruction text prepended to the NLI. The prefix we used was “*Paraphrase the following instruction while maintaining semantics and hiding authorship. Return only the paraphrased output, nothing else*”.

Recent evidence shows that giving an LLM a single high-level instruction such as ‘Paraphrase the following text to obfuscate the author’s identity while maintaining the meaning. Only return the paraphrased text.’ can sharply reduce authorship-verification performance [296]. On standard benchmark corpora, the F_1 score drops from ≈ 0.90 to ≈ 0.41 with Llama3 and to ≈ 0.54 with GPT-4. This represents nearly a 40-point decline, yet semantic fidelity remains intact. Consequently, the brief phrase ‘hiding authorship’ in our fixed prompt has precedent in adversarial stylometry research and appears reasonably precise for LLMs to yield effective obfuscations, as indicated by our results in Table 6.8

and Figure 6.10.

Static and Adaptive Adversary. To be more rigorous in our evaluation of the defense mechanisms, we need to consider an adversary that is in knowledge of the defense mechanisms and has the ability to adopt to it. Thus we consider both a *static* and an *adaptive* version of the $\mathcal{A}_i$ adversary. In a static or non-adaptive setting, the adversarial models are trained on the source data (similar to the models presented this far) and tested on obfuscated data. In the adaptive setting, the adversarial models are trained on both the original and the obfuscated data by the respective mechanism and tested on unseen obfuscated data. We select our best performing $\mathcal{A}_i$ model, *InferSent*, and run it both in a static and an adaptive setting when each defense mechanism is in place.

6.5.2 Results

Next, we examine whether these techniques offer a good trade-off between authors' privacy and semantics.

Privacy. The privacy results of the static and adaptive adversary against the mitigation strategies are summarized in Figure 6.10 where we show the adversary's performance under a limited sample setting with $10 \leq |D_{aux}^\alpha| \leq 80$. Most mitigation strategies seem promising against a static adversary which stays below a 60% Top-5 accuracy with 80 training samples. In particular, when leveraging $|D_{aux}^\alpha| = 70$ training samples the Top-5 accuracy drops to 50% and lower (see Figure 6.10) for both LLMs and for Pegasus with $\epsilon \leq 0.33$. Llama3 offers better protection than Mistral as Top-5 attack accuracy against Llama3 drops to 34% compared to 50% against Mistral.

Under the more realistic *adaptive adversary*, we observe that both Mistral and Llama3 fail to maintain similar levels of privacy protection. Mistral allows the attacker to reach 70% Top-5 accuracy with $|D_{aux}^\alpha| = 70$ training samples. Llama3 offers better protection compared to Mistral with the attack success rate dropping to 63% Top-5 accuracy but

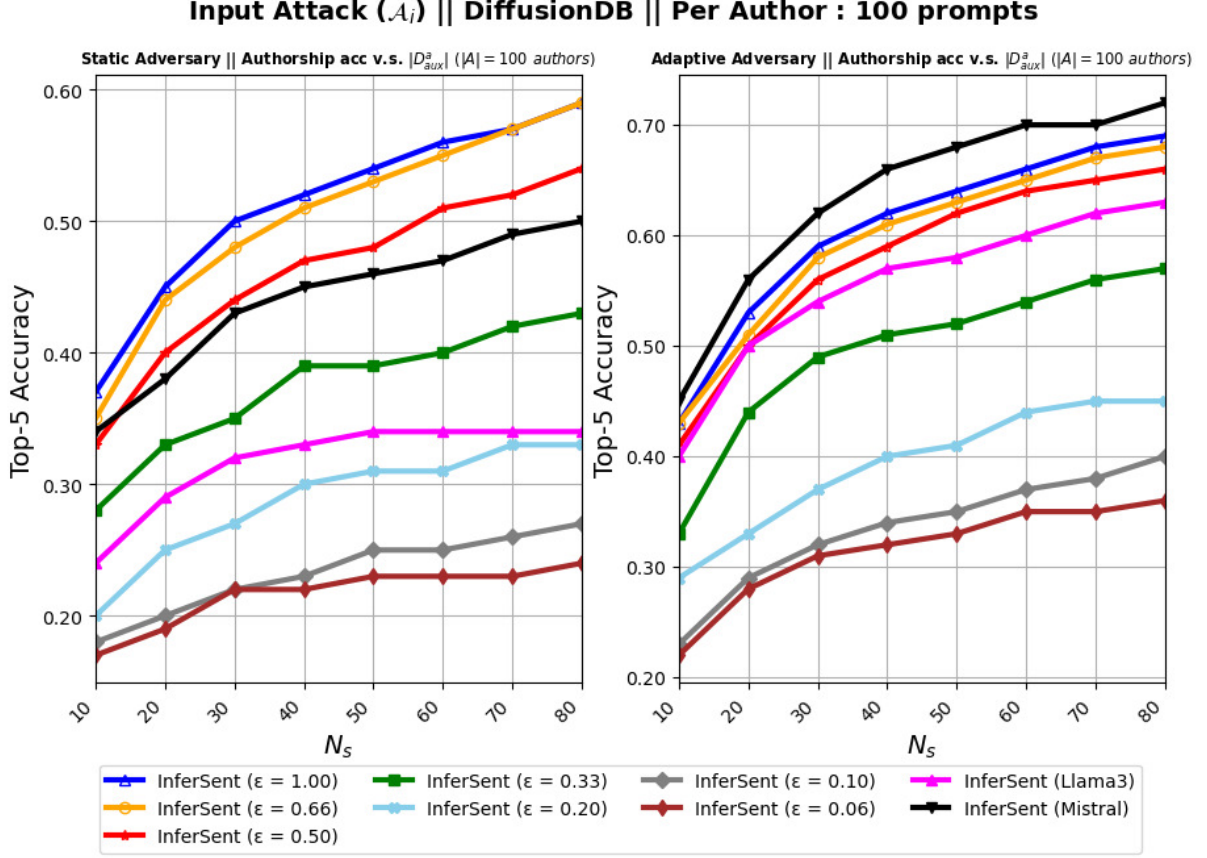


Figure 6.10: The attack success rate of static and adaptive adversarial InferSent-based models against LLMs and PEGASUS. The defense method is shown in parentheses.

much worse than its own performance against a static adversary. We argue that this happens because Mistral preserves higher NLI semantics (as we show in the utility analysis). Pegasus with low privacy protection $\epsilon = 1, 0.66, 0.50$ also fails to offer good guarantees against the adaptive adversary (Top-5 accuracy $> 64\%$ on 70 training samples and Top-5 accuracy $> 66\%$ on 80 training samples). As we increase the privacy protection for Pegasus (lower ϵ) the adversary success drops below 60% but with increased ϵ there is also a bigger penalty on utility as we will analyse later on. Overall, Llama3 and Pegasus ($\epsilon \leq 0.33$) are the most promising in providing some privacy protection.

Utility of NLIs. To examine the defense' ability to preserve utility we first look at the input space (text) and how the each technique affects the semantics of the original NLIs. To measure this, we compute the cosine similarity of the original NLIs with paraphrased

ones using the *all-MPNet-Base-v2* sentence transformers [264] model. The results for all potential mitigation strategies are summarized in Table 6.8.

Table 6.8: Semantic similarity between original and paraphrased NLIs, using PEGASUS, Llama3 and Mistral.

Sequence-to-Sequence Model (PEGASUS)							
Temperature (T)	1.0 (No DP)	1.5	2.0	3.0	5.0	10.0	15.0
Privacy Budget ($\epsilon = 1/T$)	1.0	0.67	0.50	0.33	0.20	0.10	0.07
Mean Similarity	0.770	0.757	0.730	0.670	0.588	0.537	0.520
Large Language Models (LLMs)							
	Llama3 (8B)			Mistral (7B)			
Mean Similarity	0.693			0.743			

We observe that as expected, the more we increase the privacy protection for Pegasus (lower ϵ) the lower the similarity (worse semantics) is between the original NLI and the paraphrased one. Mistral achieves the higher similarity but as we have seen the privacy protection offered is limited.

Overall, the most promising approach is Pegasus with $\epsilon = 0.33$. This achieves the best privacy-utility trade-off with Top-5 accuracy for the adaptive adversary below 60% and semantic similarity at 67%. Llama3 is close with slightly better average NLI semantic similarity but worse protection.

End-to-End Utility. To better understand how the input semantic similarity offered by promising defense mechanisms translates to the end-to-end utility of T2I models we first perform a quantitative analysis comparing the inputs with the resulting outputs and then provide a qualitative example.

To study whether the generated images align with the intentions of the original NLIs, we compare the original (not obfuscated) NLI with the output of the T2I model given the *obfuscated* NLI. This simulates the scenario where the user provides their original NLI, and then a privacy layer obfuscates the NLI before making it available to the T2I model and more broadly accessible to the adversary. We envision this layer to be part of a user’s device (mobile phone, browser, or voice assistant). To compare the text input with the

image output we use CLIP score computed as described in Section 6.4.3.

For our analysis we chose the two most promising mitigation strategies Llama3 and Pegasus with $\epsilon = 0.33$. As a baseline for comparison we also add PEGASUS without DP ($\epsilon = 1.0$) which is one of the state of the art paraphrasing models and therefore while not optimised for privacy it yields good semantics as it is also shown on Table 6.8. Lastly we also add the original CLIP score of the target model which can serve as a baseline of end-to-end utility without any defense. For the experiment, and for each case, we randomly select 10 paraphrased NLIs of 10 authors. The authors are randomly sampled from the 100 authors dataset used in the Input-Output Attack ($\mathcal{A}_{io}$) (see Section 6.4.1). For each NLI we generate 5 sample images with Stable Diffusion v2.1. This resulted in 500 new images per model in total. Next, we compute the CLIP scores between the *original* NLIs and the synthetic images generated from the paraphrased NLIs. Figure 6.11 presents the CDF of the CLIP scores with different methods. We observe that only a small percentage of the generated images of all defense models has ≥ 0.26 CLIP score, indicating a large loss in semantics in the output space. Hence, none of the mitigation techniques provide a sufficient trade-off between the users' privacy and semantics in both input and output space.

Lastly, we provide an example of how an input NLI is paraphrased by different defense mechanisms and the resulting generated image from the paraphrased NLI in Figure 6.12. We observe that when $\epsilon = 0.33$ (Figure 6.12f) important concepts from the original image are missing (e.g., samurai's head is omitted).

Overall. Our analysis showed that text paraphrasing can be a promising approach to obfuscate authorship. Techniques like LLMs (Llama3) and word differential privacy with Pegasus can offer some protection, but the loss in semantics might not be acceptable and thus more needs to be done to adequately and realistically protect against the T2I authorship inference adversary.

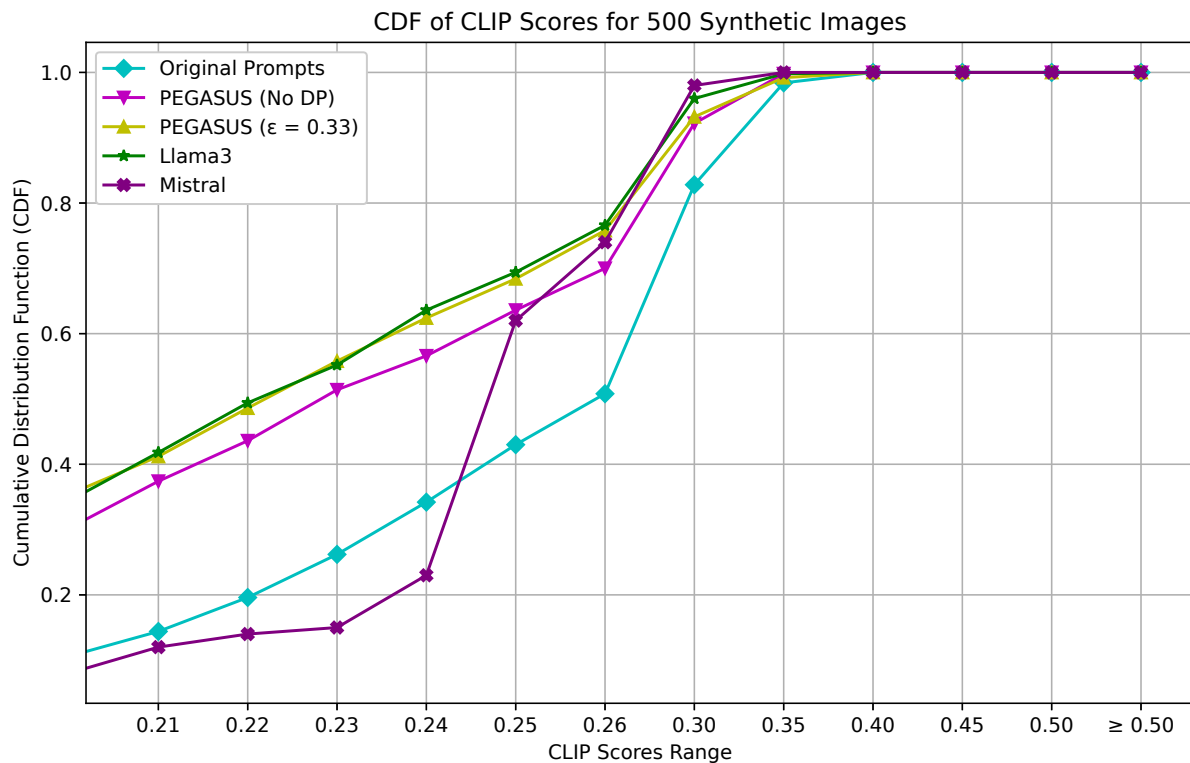


Figure 6.11: CDF of CLIP scores for 500 images generated from original NLIs (Stable Diffusion v2.1), and from paraphrased NLIs using different obfuscation mechanisms.

6.5.3 Directions for Improved Mitigation

Our experiments suggest that while current text paraphrasing methods—whether using LLMs or adjusting sampling temperatures—can offer some degree of privacy, it often distorts the prompts’ intended meaning, resulting in less semantically close generated images. As a result, these methods are insufficient to achieve a satisfactory trade-off between semantic fidelity and author’s anonymity. Looking ahead, a natural next step involves the design and exploration of *hybrid* approaches that not only hide stylistic fingerprints but also retain users’ intent. Below, we discuss several potential mitigation directions for more robust and practical solutions.

Stylometric Alteration. Beyond simple paraphrasing, to better mask an author’s unique style, we can incorporate more advanced stylometric techniques [305, 40, 151]. For instance, by i) altering sentence structure, ii) normalizing punctuations, and iii) applying



(a) samurai mecha on ghost of tsushima box art (Original DiffusionDB image)



(b) samurai mecha on ghost of tsushima box art (Image from Stable Diffusion v2.1 using the original description)



(c) The ghost of tsushima box art has samurai mecha on it. (Similarity score: 0.88 with $T = 1$)



(d) The samurai mecha is on the ghost of tsushima box art. (Similarity score: 0.88 with $T = 1.5$)



(e) The samurai mecha is on the ghost of shushima box art (Similarity score: 0.85 with $T = 2$).



(f) samurai mecha (Similarity score: 0.67 with shushima box art (Similarity score: 0.85 with $T = 3$).



(g) There were samurai mehaga on that piece (Similarity score: 0.59 with $T = 5$).



(h) There are art objects for which samurai iconography can be associated - samurai me (Similarity score: 0.58 with $T = 10$).



(i) There are art reproductions of Tokugawa box pieces. benkins on (Similarity: 0.54 with $T = 15$).



(j) A cover of Ghost of Tsushima features a striking fusion of traditional Japanese warrior attire and futuristic machinery, in a unique and captivating image of a samurai-inspired mecha. (Similarity score: 0.78 with Llama3)



(k) Mecha samurai depicted on the cover art of Ghost of Tsushima (Similarity score: 0.88 with Mistral).

Figure 6.12: Qualitative comparison of end-to-end utility of mitigation strategies. T2I images, their NLI and text similarity score in parenthesis with the original NLI. (a) shows the original image included in DiffusionDB; (b) the original NLI is given as input to Stable Diffusion v2.1 [268]; in (c), (d), (e), (f), (g), (h) and (i) the input is the paraphrased NLI with different temperatures ($T = 1, 1.5, 2, 3, 5, 10$ and 15). In (j) and (k) we use the paraphrases from Llama3 and Mistral. The full text produced by Llama (j) is “A cover of Ghost of Tsushima features a striking fusion of traditional Japanese warrior attire and futuristic machinery, in a unique and captivating image of a samurai-inspired mecha.”

lexical substitution schemes can further reduce the distinct patterns that adversaries rely on [40, 13]. Therefore, such targeted linguistic changes can offer improved privacy without sacrificing semantic content.

Regularization and Model Retraining. Relying solely on post-hoc transformations of user prompts may be inadequate. Future work can focus on embedding privacy-preserving mechanisms directly into the training and fine tuning of the T2I models. For example, by training image generation models with specific regularisation strategies, we can discourage them from becoming too attuned to individual writing styles. Hence, we can reduce the model’s susceptibility to authorship cues [111, 206]. Furthermore, methods inspired by domain generalization and adversarial training [186, 103] can be used to guide the model toward ignoring distinctive stylistic indicators. Such built-in safeguards can make it harder for adversaries to trace prompts back to the original authors.

Input–Output Decoupling and Specialized Prompt Tuning. Addressing obfuscation separately in the input (prompt) and output (generated image) space can further improve authors’ privacy. On the input side, prompt-tuning (e.g., prefix-tuning [190, 181], or p-tuning [194, 195]) can systematically introduce subtle and controlled stylistic noise without losing the prompt’s original intent. On the output side, post-processing steps—such as applying style transfer techniques, image filters, or mild perturbations—can mask distinctive visual signatures. By coordinating these input and output modifications carefully, we can help generated images remain semantically relevant.

User–Centric Privacy Customization. Ultimately, privacy protection should empower users with fine-grained control over their desired trade-off between utility and privacy. Rather than relying on a one-size-fits-all solution, interactive interfaces should allow users to specify their tolerance for semantic shift; choosing how much ‘noise’ they are willing to accept. For example, an author might opt for minimal paraphrasing (low privacy) if his primary concern is semantic fidelity, while another author might prefer stronger stylistic transformations (high privacy) at the cost of reduced coherence. By

tailoring these options to the authors’ preferences, solutions can accommodate a broad variety of use cases. More recently, advances in LLM prompting techniques [18] suggest that users can now more effectively guide these models through carefully crafted prompt instructions. As a result, by combining such adaptive prompting strategies with user-defined stylometric adjustments, it may be possible to achieve a finer level of control.

Overall. Future mitigations will likely leverage a combination of refined stylometric changes. Our foundational insights and challenges highlighted in this research, further studies can more effectively balance the goal of protecting user privacy with the need to keep T2I systems practical and useful.

6.5.4 Conclusion

We have revealed the threat of attribute leakage in text-guided diffusion-based image generation models. Unique to our work we model a hierarchical adversary with different access levels on the input, output or multi-modal input-output access to such models. We show that by leveraging data augmentation and models which learn joint representations of text and images to extract multi-modal embeddings adversaries can achieve state of the art results in spear authorship inference even with limited training data. Our inference models are rigorously evaluated and demonstrated robustness against different training datasets and diffusion models, number of authors, and strong mitigation strategies. We have also shown that text-image encoders used to guide diffusion models encode pathological speech markers which allows an adversary to infer the presence of dementia from natural language instructions and generated images, highlighting a previously unidentified threat for a vulnerable understudied population whose numbers are rapidly increasing. We hope our work raises awareness on the privacy hazards on the usage of diffusion-based image generation models, and inspire further work on privacy leakages in such models and mitigation mechanisms to warrant users’ privacy while achieving acceptable model utility.

Ethics. We analyzed data from *DiffusionDB* dataset [342] and demonstrated that NLI

and/or the generated images lead to authorship attribution. We note that all generated images have a CC0 1.0 license that allows uses for any purpose [5]. We did not extract any personal information linked to individuals as this could potentially be sensitive. We also a) never share any information with 3rd parties and b) encrypt and keep in a local disk all the data after each experiment. We furthermore follow recommendations in [349] and we do not include natural language instructions of individual authors.

Chapter 7

Conclusion

This final chapter revisits the contributions, achievements and limitations of this thesis. Future research directions are also discussed, and we attempt to assess the potential scientific impact of this thesis.

7.1 Summary of Thesis Achievements

This thesis makes foundational contributions to the study of online identity vulnerabilities by addressing both explicit impersonation and implicit inference attacks. In particular, it (a) identifies a novel strategy employed by adversaries to impersonate users via squatted usernames on online social networks (OSNs); (b) develops innovative adversarial pipelines for inferring authorship and sensitive attributes from multimodal content; and (c) introduces tools and frameworks that assist model developers and platform providers in the detection and mitigation of these threats.

The work presented in this thesis is build upon the following pillars:

Motivation : In today’s digital era, online identities have become integral to personal,

professional, and societal interactions. These identities—whether explicitly defined via usernames or subtly encoded in user-generated content—enable trust, attribution, and continuity across online platforms. However, this reliance has also rendered them vulnerable to exploitation. Malicious actors now exploit not only visible identity markers, but also latent cues embedded in AI-generated content. This development underscores the urgent need to re-evaluate privacy and authenticity standards in the digital domain.

This research was motivated by the escalating threat of identity misuse. Firstly, online impersonation via username squatting has emerged as a pervasive and socially disruptive phenomenon, targeting high-profile accounts to deceive, manipulate, and amplify misinformation. Secondly, advancements in generative machine learning—particularly text-to-image diffusion models—have introduced subtle yet significant risks, whereby seemingly benign outputs can inadvertently reveal sensitive personal attributes, such as authorship or health status.

Collectively, these dual threads point to a broader challenge: as online identities become more complex and distributed across multiple modalities, they expose new attack surfaces that require both detection and mitigation.

In sum, the motivation behind this work is rooted in addressing these key objectives:

- **Safeguarding Online Identities** : Ensuring that individuals and organizations can preserve their authenticity and credibility in the digital space without falling victim to impersonation.
- **Strengthening Trust in OSNs** : Uncovering the mechanisms and impact of username squatting to guide platform-level interventions that enhance user safety and reduce identity confusion.
- **Advancing the Understanding of Attribution Risks** : Demonstrating how adversarial inference pipelines can extract sensitive traits—such as authorship or dementia status—from multimodal data, even in black-box generative settings.

SQUAD. Our work is the first which studies *username squatting* on OSNs. We define and characterize *username squatting*, and develop an efficient and effective methodology for identifying squatted accounts on OSNs. To achieve our first goal, we design a systematic measurement study for one of the most popular networks [236], *X* (previously known as Twitter [328]). Our study is structured around four measurement questions: **(MQ1)** Is *username squatting* a prevalent problem on *X*? If it is, then **(MQ2)** how does it contribute to online confusion; **(MQ3)** what are the username characteristics of the most problematic username variants, and **(MQ4)** what is the specific behaviour of the squatted accounts (i.e., what is the *purpose* of the owner of such accounts)?

To aid our analysis, we developed a tool, UsernameCrazy, inspired by typo- and mobile app- squatting methodologies [336, 139] but tailored to the characteristics of OSN accounts, which uses 10 string generation strategies to efficiently construct more than 851,000 username variants for the top 97 (in 2019) popular *X* accounts [243], for which we collected more than 1.2 million tweets in total. Our measurement study led to a number of important findings. We found that tens of thousands of potentially confusing variants exist on the network; a large number of username variants of popular accounts are *mistakenly* mentioned (typo-mentioned) and that confusing variants appear high in the OSN’s search recommendation list—and at times even *before* the verified account (see Section 4.4.2). We found that variants with a small edit distance are more likely to be mentioned and suggested. Both phenomena lead to increased attention for these confusing accounts which malicious actors can consequently exploit. As proof of this, we observed that a subset of these variants is already suspended, which suggests that *username squatting* is indeed leveraged for malicious purposes. Moreover, our three data collection snapshots across one year show that this is a growing problem, as we found the number of suspended variants to increase over time. Analysing the squatted accounts, we show that thousands of squatted accounts which are currently *active* on the network are likely bots. A significant amount is involved in selling products, political goals or spreading fake news. Thus, as expected, the confusion generated through similar usernames and

profiles tends to be exploited with malicious intentions.

Our measurement study shows that many squatted username accounts are involved in previously unrecognized impersonation attacks. Furthermore, our findings reveal that profiles sharing many similar features with the original account may try to exploit the created confusion and generate traction to their profile to promote products or services, or share potentially malicious links. This led us to our design question: **(DQ)** Can we design an effective and efficient framework for detecting suspicious username squatting attempts? Toward this, we leverage our previous observations to design **SQUAD**. SQUAD combines user name generation and online account discovery with the classification of suspicious squatted accounts. By ensuring a low false positive rate, we envision our framework to be an effective tool in scanning the OSN, saving time in manual screening and resulting in the more rapid discovery of suspicious profiles. A prototype of SQUAD trained on a small dataset reaches an F1-score of 94% in detecting suspicious accounts. We further applied SQUAD on popular and non-popular accounts and found that squatting targets popular users (55% of accounts squatting popular users are classified as malicious). Upon analysing their activity we reveal two trends: (a) posting insecure URLs and at a lesser but non-negligible extend (b) trying to amplify their follower base. Examples of impersonators, confusing variants, fake tweets and suspicious links, and additional results are available on our project’s website [175].

Authorship and Attribute Inference. Extending the notion of identity beyond user-chosen names, this thesis also investigates implicit identity leakage from generative AI systems. We demonstrate that text-to-image diffusion models—despite being trained for visual creativity—can inadvertently encode and expose sensitive user attributes. By designing adversarial pipelines that combine prompt manipulation, image augmentation, and multimodal embedding models, we achieved high accuracy in inferring authorship and medical conditions (e.g., dementia) from model outputs. These attacks operate under black-box settings, indicating systemic vulnerabilities independent of training data specifics or classifier choice.

A critical contribution of this work lies in its evaluation of state-of-the-art privacy-preserving techniques. Our results underscore a fundamental trade-off: although certain defenses can reduce inference success, they often come at the cost of degraded content utility. This finding highlights a core tension in generative AI—balancing privacy and functionality—which must be addressed for future systems to be both safe and effective.

Together, these results underscore the need for robust, generalizable mitigation strategies to protect users from unintended leakage in multimodal generative environments. They expand the definition of online identity threats to include not only overt impersonation, but also covert inference of personal characteristics from AI-generated content.

Code availability. The contributions of this thesis (code implementation, dissemination, experiments) are publicly available. The code integrated into UsernameCrazy, SQUAD, and inference attacks are publicly available via MIT Licence. The hyperlinks to reproduce this work are available in their respective chapter. See subsection 4.2.4 for Chapter 4, subsection 5.5.3 for Chapter 5 and subsection 6.4.5 for Chapter 6.

7.1.1 Future Research Directions

Despite its significant contributions, the research presented in this thesis is subject to several limitations that present opportunities for further investigation. This subsection will elaborate on these limitations and outline prospective research directions aimed at addressing them. Some of the limitations of the research presented in this thesis have already been partially discussed in the previous chapters (see Section 5.4.2 for Chapter 5 and Section 6.5 for Chapter 6). In this subsection, we revisit these points, build upon previously mentioned mitigations, and introduce additional strategies to address them more effectively.

UsernameCrazy : While UsernameCrazy generates a significant large set of username variants, it does not exhaustively explore every possible combination. This limitation

leaves gaps that could potentially be exploited by adversaries using alternative squatting methods not covered by the current version of UsernameCrazy. Moving forward, UsernameCrazy should aim to refine and expand the variety and completeness of the generated username variants by incorporating machine learning-based generative models, such as GPT-based sequence generators, to dynamically produce novel variations beyond predefined patterns, while maintaining its computational efficiency. Moreover, by integrating adaptive algorithms that respond to emerging squatting patterns could significantly enhance tool's robustness and help it evolve alongside adversarial tactics. Such algorithms could leverage reinforcement learning to prioritize likely future patterns. Rigorous validation methods must also be implemented in order to ensure that the produced variants remain contextually relevant and align with the username policies of the platforms.

SQUAD Evaluation : Drawing from the insights presented in Section 5.4.2, the dataset used for training and evaluation of SQUAD is relatively small. More specifically, the whole dataset comprises 1,378 users, thereby constraining the diversity of online behaviours and account types that can be represented. Collaborating with OSNs could facilitate access to real-world datasets enhancing the frameworks' applicability. Additionally, incorporating synthetic datasets that can simulate emerging impersonations and other squatting behaviours could provide broader evaluation. Ensuring that datasets reflect more diverse user demographics, including cultural, linguistic and geographic variations, will be essential to improve generalisability.

Photo Similarity : The current image similarity algorithm is limited to facial comparisons. Extending its capabilities to include the analysis of non-facial images could increase its versatility and effectiveness in addressing more diverse online impersonation scenarios. For instance, an impersonator targeting a public figure like Barack Obama might use a profile photo of the White House. Incorporating more advanced image analysis algorithms to include such contexts could significantly improve impersonation detection accuracy. Building on this premise, it is crucial to acknowledge that non-facial images can also act as powerful 'proxies' for identity in social platforms. The sophistication

of such systems relies on their ability to perform high-level semantic analysis. Beyond basic pixel similarity, advanced approaches could utilize CNNs or ViTs to recognise symbolic references—such as landmark buildings, logos or other signature elements—commonly related to specific individuals, public offices, institutions or brands.

Verification Badge. Our measurement study showed that large amount of squatted username accounts are involved in *previously unknown* impersonation attacks; see for an example Figure 4.2a. To tackle such issues, *X* recently launched a premium service which allowed everyone paying a small fee (\$8 per month) to be granted a blue check on their account [325]. Blue checks signal account verification and this feature was quickly and heavily manipulated by a swath of fake accounts to spread misinformation [146] and even damage companies’ brands and stock prices [104, 71]. Companies reported to be withdrawing their ad campaigns from *X* due to the uncertainty of the platform’s directions and to be pausing their *X* publishing plans for all their corporate accounts [123, 262, 83]. The premium feature was reverted and *X* claimed that blue checks will be granted only after manual verification [346]. The platform relaunched a modified version of their premium service which uses color-coded badges for different entities (organisations vs individuals etc.) along with an increased fee (\$11 per month) if registered from a mobile device [325]. However, anecdotal evidence suggest that color badges are not a foolproof method, allowing adversaries to add cloned badges next to their profile name [134, 56]. Also, manual verification is cumbersome and costly which is further exacerbated by the fact that OSNs currently operate with restricted resources [57, 58].

Black-Box Inference Models: The evaluation performed in Chapter 6 relied on a limited set of available prompts per author. This might pose a constraint on the generalizability of the results. While the current evaluation highlights the potential for authorship and attribute leakage, the small dataset size can restrict a comprehensive understanding of the models’ behaviours on a larger number of authors. Therefore, a natural next step is to focus on larger number of authors (see Figure 6.4) and more diverse prompts, in order to cover a broader range of topics, styles as well as contexts.

Mitigation of Attribute Leakage: Building upon the findings and suggestion in Section 6.5.3, future work should develop robust defences against adversarial attacks targeting T2I models. Techniques leveraging adversarial training and reinforcement learning could offer promising avenues for enhancing privacy without degrading utility. Furthermore, the integration of multimodal datasets into obfuscation frameworks presents an exciting frontier for advancing both authorship attribution and privacy-preserving technologies.

Open-World Adaptation – Discussion: In many real-world deployments an adversary cannot enumerate all possible participants, so prompts from previously unseen authors inevitably arise. This *open-world* setting—common on public discussion forums and social-network groups—was investigated, *focusing exclusively on the input-attack model*, in an experiment led by my co-author *Le Thanh Hai*. The full protocol and raw result tables are available in our paper “*Leaky Diffusion: Attribute Leakage in Text-Guided Image Generation*” (in Sections 3.5 and 5.4 of the paper); they are omitted here to avoid redundancy and to keep the thesis centred on the closed-world contribution. In brief, even when unseen authors outnumbered target authors by 90:1, our inference model A_i achieved Top-1 and Top-5 attribution accuracies of 62% and 69%, respectively, demonstrating robustness and scalability in open-world conditions and underscoring the framework’s practical applicability. The examination of the output-only (A_o) and combined input-output (A_{io}) attack models under open-world assumptions is left for future exploration.

Advanced Research Directions : Incorporating explainable AI (XAI) methods into our detection frameworks can significantly enhance transparency and trust among users and stakeholders. Extending the analysis of squatted accounts to include behavioural patterns, such as posting frequency, engagement rates and network dynamics with further refine detection capabilities of online impersonation. XAI methods can also help to better explain why a classifier is able to distinguish authors from text, images, or both, providing valuable insights into the underlying differences in authors’ styles or attributes. This understanding will improve the interpretability of classification results and support the

refinement of models to address specific authorship inference challenges. Consequently, these advanced directions will not only enhance the robustness of current solutions but also pave the way for more comprehensive and adaptive frameworks.

Addressing the outlined limitations and future directions will further advance the field of online identity protection and content attribution.

Bibliography

- [1] “California consumer privacy act of 2018,” Cal. Civ. Code §1798.100 et seq., 2018.
- [2] “Meta - llama3,” <https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct>, 2024, accessed: 2024-03-01.
- [3] “Mistral,” <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3>, 2024, accessed: 2024-03-01.
- [4] D.-E. 2, “Dall-e 2 subreddit.” <https://www.reddit.com/r/dalle2/>, accessed: 2024-07-11.
- [5] S. 2022b., “Terms of use.” <https://stability.ai/terms-of-use>, 2022.
- [6] J. Aas. (2015) Let’s encrypt: The ca’s role in fighting phishing and malware. Accessed: 2023-06-01. [Online]. Available: <https://letsencrypt.org/2015/10/29/phishing-and-malware.html>
- [7] B. Acharya, D. Lazzaro, E. López-Morales, A. Oest, M. Saad, A. E. Cinà, L. Schönherr, and T. Holz, “The imitation game: Exploring brand impersonation attacks on social media platforms,” in *33rd USENIX Security Symposium (USENIX Security 24)*, 2024, pp. 4427–4444.
- [8] N. ACOMI, O. ACOMI, E. R. ANDREI, F. DADOMO, D. HOCQ, O. S. Luis, R. OCHOA-DADERSKA, F. PORCELLI, S. RICCHIUTO, and A. VE-

- LASCO GARCIA, “Empowering youth to critically analyse fake news strategies of intervention and good practices,” 2019.
- [9] A. Acquisti, C. Taylor, and L. Wagman, “The economics of privacy,” *Journal of economic Literature*, vol. 54, no. 2, pp. 442–492, 2016.
- [10] I. agency, “Federal trade commission,” <https://www.ftc.gov/>, accessed: 2023-12-01.
- [11] P. Agten, W. Joosen, F. Piessens, and N. Nikiforakis, “Seven months’ worth of mistakes: A longitudinal study of typosquatting abuse,” in *NDSS*, 01 2015.
- [12] D. Al-Khoury, “What is harmful content?” *The Routledge Handbook of Conflict and Peace Communication*, 2024.
- [13] M. Almishari, E. Oguz, and G. Tsudik, “Fighting authorship linkability with crowdsourcing,” in *Proceedings of the second ACM conference on Online social networks*, 2014, pp. 69–82.
- [14] F. Amato, A. Castiglione, A. D. Santo, V. Moscato, A. Picariello, F. Persia, and G. Sperl , “Recognizing human behaviours in online social networks,” *Comput. Secur.*, vol. 74, pp. 355–370, 2018.
- [15] Apify, “Technologies,” <https://apify.com/stifl/reddit-username-from-subreddit>, accessed: 2024-07-11.
- [16] S. Argamon and S. Levitan, “Measuring the usefulness of function words for authorship attribution,” in *Proceedings of the 2005 ACH/ALLC Conference*, 2005, pp. 4–7.
- [17] A. Arora, P. Nakov, M. Hardalov, S. M. Sarwar, V. Nayak, Y. Dinkov, D. Zlatkova, K. Dent, A. Bhatawdekar, G. Bouchard *et al.*, “Detecting harmful content on online platforms: what platforms need vs. where research efforts go,” *ACM Computing Surveys*, vol. 56, no. 3, pp. 1–17, 2023.

- [18] S. Arora, A. Narayan, M. F. Chen, L. Orr, N. Guha, K. Bhatia, I. Chami, F. Sala, and C. Ré, “Ask me anything: A simple strategy for prompting language models,” *arXiv preprint arXiv:2210.02441*, 2022.
- [19] H. Askari, A. Chhabra, M. Chen, and P. Mohapatra, “Assessing llms for zero-shot abstractive summarization through the lens of relevance paraphrasing,” *arXiv preprint arXiv:2406.03993*, 2024.
- [20] Z. Ba, J. Zhong, J. Lei, P. Cheng, Q. Wang, Z. Qin, Z. Wang, and K. Ren, “Surrogateprompt: Bypassing the safety filter of text-to-image models via substitution,” *arXiv preprint arXiv:2309.14122*, 2023.
- [21] A. Badawy, A. Addawood, K. Lerman, and E. Ferrara, “Characterizing the 2016 russian ira influence campaign,” *Social Network Analysis and Mining*, vol. 9, pp. 1–11, 2019.
- [22] Badiuca, “Art in protest,” <https://artinprotest.viewingrooms.com/viewing-room/9-badiuca/>, 2024, accessed: 2024-07-26.
- [23] B. BAKHSHANDEH, “Kaggle spammer dataset,” <https://www.kaggle.com/free4ever1/instagram-fake-spammer-genuine-accounts>, accessed: 2021-10-14.
- [24] S. Bakhshi, D. A. Shamma, and E. Gilbert, “Faces engage us: Photos with faces attract more likes and comments on instagram,” in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2014, pp. 965–974.
- [25] A. Balagopalan, B. Eyre, F. Rudzicz, and J. Novikova, “To bert or not to bert: comparing speech and language-based approaches for alzheimer’s disease detection,” *arXiv preprint arXiv:2008.01551*, 2020.
- [26] M. Banko, B. MacKeen, and L. Ray, “A unified taxonomy of harmful content,” in *Proceedings of the fourth workshop on online abuse and harms*, 2020, pp. 125–137.

- [27] M. Barz, S. Stauden, and D. Sonntag, “Visual search target inference in natural interaction settings with machine learning,” in *ACM Symposium on Eye Tracking Research and Applications*, 2020, pp. 1–8.
- [28] J. T. Becker, F. Boiler, O. L. Lopez, J. Saxton, and K. L. McGonigle, “The natural history of alzheimer’s disease: description of study cohort and accuracy of diagnosis,” *Archives of neurology*, vol. 51, no. 6, pp. 585–594, 1994.
- [29] W. Benedek and M. C. Kettemann, *Freedom of expression and the internet: Updated and revised 2nd edition*. Council of Europe, 2020.
- [30] Y. Bengio, R. Ducharme, and P. Vincent, “A neural probabilistic language model,” *Advances in neural information processing systems*, vol. 13, 2000.
- [31] L. Berrada, A. Zisserman, and M. Kumar, “Smooth loss functions for deep top-k classification,” *arXiv preprint arXiv:1802.07595*, 02 2018.
- [32] C. Besel, J. Echeverria, and S. Zhou, “Full cycle analysis of a large-scale botnet attack on twitter,” in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018, pp. 170–177.
- [33] J. Bevendorff, M. Potthast, M. Hagen, and B. Stein, “Heuristic search-based authorship obfuscation,” in *Proceedings of the 2019 Conference on Computational Linguistics*, 2019.
- [34] L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda, “All your contacts are belong to us: Automated identity theft attacks on social networks,” in *Proceedings of the 18th international conference on World wide web*, 01 2009, pp. 551–560.
- [35] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O’Reilly Media, Inc.", 2009.
- [36] T. Bohm, “Letter and symbol misrecognition in highly legible typefaces for general, children, dyslexic, visually impaired and ageing readers,” *Information Design Journal*, vol. 21, 12 2014.

- [37] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the association for computational linguistics*, vol. 5, pp. 135–146, 2017.
- [38] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, “The socialbot network: when bots socialize for fame and money,” in *Proceedings of the 27th annual computer security applications conference*, 2011, pp. 93–102.
- [39] D. M. Boyd and N. B. Ellison, “Social network sites: Definition, history, and scholarship,” *Journal of computer-mediated Communication*, vol. 13, no. 1, pp. 210–230, 2007.
- [40] M. Brennan, S. Afroz, and R. Greenstadt, “Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity,” *ACM Transactions on Information and System Security (TISSEC)*, vol. 15, no. 3, pp. 1–22, 2012.
- [41] G. Broussalis, G. Markopoulos, and G. Mikros, “Stylometric profiling of the greek legal corpus,” in *Selected Papers of the 10th International Conference of Greek Linguistics*, vol. 10, 2012, pp. 167–176.
- [42] B. Byeon and K. Rasheed, “Simultaneously removing noise and selecting relevant features for high dimensional noisy data,” in *2008 Seventh International Conference on Machine Learning and Applications*, 2008, pp. 147–152.
- [43] J. A. Campos and J. Shern, “Training language models with language feedback,” in *ACL Workshop on Learning with Natural Language Supervision. 2022.*, 2022.
- [44] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro, “Aiding the detection of fake accounts in large scale social online services,” in *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, ser. NSDI’12. USA: USENIX Association, 2012, p. 15.

- [45] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, “Vggface2: A dataset for recognising faces across pose and age,” in *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, 2018, pp. 67–74.
- [46] N. Carlini, J. Hayes, M. Nasr, M. Jagielski, V. Sehwag, F. Tramèr, B. Balle, D. Ippolito, and E. Wallace, “Extracting training data from diffusion models,” in *Proceedings of the 32nd USENIX Conference on Security Symposium*, ser. SEC ’23. USA: USENIX Association, 2023.
- [47] N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson *et al.*, “Extracting training data from large language models,” in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 2633–2650.
- [48] N. Chavoshi, H. Hamooni, and A. Mueen, “Temporal patterns in bot activities,” in *Proceedings of the 26th international conference on world wide web companion*, ser. WWW ’17 Companion. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2017, pp. 1601–1606. [Online]. Available: <https://doi.org/10.1145/3041021.3051114>
- [49] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, “Smote: Synthetic minority over-sampling technique,” *J. Artif. Intell. Res. (JAIR)*, vol. 16, pp. 321–357, 06 2002.
- [50] G. H. Chen, S. Chen, Z. Liu, F. Jiang, and B. Wang, “Humans or llms as the judge? a study on judgement biases, 2024,” *URL <https://arxiv.org/abs/2402>*, vol. 10669.
- [51] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy, “Make new friends, but keep the old: recommending people on social networking sites,” in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2009, pp. 201–210.

- [52] X.-w. Chen and J. C. Jeong, “Enhanced recursive feature elimination,” in *Sixth International Conference on Machine Learning and Applications (ICMLA 2007)*, 2007, pp. 429–435.
- [53] B. Chesney and D. Citron, “Deep fakes: A looming challenge for privacy, democracy, and national security,” *Calif. L. Rev.*, vol. 107, p. 1753, 2019.
- [54] K. Chlasta and K. Wolk, “Towards computer-based automated screening of dementia through spontaneous speech,” *Frontiers in Psychology*, vol. 11, p. 623237, 2021.
- [55] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, “Detecting automation of twitter accounts: Are you a human, bot, or cyborg?” *IEEE Transactions on dependable and secure computing*, vol. 9, no. 6, pp. 811–824, 2012.
- [56] G. Cluley, “How twitter users can fake a verified account,” <https://nakedsecurity.sophos.com/2013/01/17/twitter-fake-verified-account/>, 2013, accessed: 2023-04-01.
- [57] CNN, “Elon musk’s twitter lays off employees across the company,” <https://edition.cnn.com/2022/11/03/tech/twitter-layoffs/index.html>, 2022, accessed: 2022-12-01.
- [58] —, “Facebook parent company meta will lay off 11,000 employees,” <https://edition.cnn.com/2022/11/09/tech/meta-facebook-layoffs/index.html>, 2022, accessed: 2022-12-01.
- [59] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [60] R. Cover, “Performing and undoing identity online: Social networking, identity theories and the incompatibility of online profiles and friendship regimes,” *Convergence*, vol. 18, no. 2, pp. 177–193, 2012.
- [61] D. R. Cox, “The regression analysis of binary sequences,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 20, no. 2, pp. 215–232, 1958.
- [62] Craiyon, “Ai image generator tool.” [Online]. Available: <https://www.craiyon.com>

- [63] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, “Fame for sale: Efficient detection of fake twitter followers,” *Decision Support Systems*, vol. 80, pp. 56–71, 2015.
- [64] —, “The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race,” in *Proceedings of the 26th international conference on world wide web companion*, 2017, pp. 963–972.
- [65] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, “Reverse stable diffusion: What prompt was used to generate this image?” *Comput. Vis. Image Underst.*, vol. 249, p. 104210, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:260438678>
- [66] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR’05)*, vol. 1. Ieee, 2005, pp. 886–893.
- [67] DALL-E, “Dall-e subreddit.” <https://www.reddit.com/r/dalle/>, accessed: 2024-07-11.
- [68] T. Dam, L. D. Klausner, D. Buhov, and S. Schrittwieser, “Large-scale analysis of pop-up scam on typosquatting urls,” in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, ser. ARES ’19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: <https://doi.org/10.1145/3339252.3340332>
- [69] F. J. Damerau, “A technique for computer detection and correction of spelling errors,” *Commun. ACM*, vol. 7, no. 3, pp. 171–176, 03 1964. [Online]. Available: <https://doi.org/10.1145/363958.363994>
- [70] G. Danezis and P. Mittal, “Sybilinfer: Detecting sybil nodes using social networks,” in *NDSS*, 2009.

- [71] A. Dangwal, ““no weapon sales to israel’: How a lockheed martin ‘tweet’ resulted in a loss of billions of dollars to us defense giant,” <https://eurasianimes.com/no-weapons-sales-to-israel-how-a-lockheed-martin-tweet-resulted/>, 2022, accessed: 2022-12-01.
- [72] J. Davis and M. Goadrich, “The relationship between precision-recall and roc curves,” in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML ’06. New York, NY, USA: Association for Computing Machinery, 2006, pp. 233–240. [Online]. Available: <https://doi.org/10.1145/1143844.1143874>
- [73] R. De Nicola, M. Petrocchi, and M. Pratelli, “On the efficacy of old features for the detection of new bots,” *Information Processing & Management*, vol. 58, p. 102685, 11 2021.
- [74] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.
- [75] —, “Bert: Pre-training of deep bidirectional transformers for language understanding,” 2019. [Online]. Available: <https://arxiv.org/abs/1810.04805>
- [76] S. Diffusion, “Stable diffusion subreddit.” <https://www.reddit.com/r/StableDiffusion/>, accessed: 2024-07-11.
- [77] A. Doan, A. Halevy, and Z. Ives, “4 - string matching,” in *Principles of Data Integration*, A. Doan, A. Halevy, and Z. Ives, Eds. Boston: Morgan Kaufmann, 2012, pp. 95–119. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780124160446000041>
- [78] J. S. Donath, “Identity and deception in the virtual community,” in *Communities in cyberspace*. Routledge, 2002, pp. 37–68.
- [79] —, “Identity and deception in the virtual community,” in *Communities in cyberspace*. Routledge, 2002, pp. 37–68.

- [80] —, “Identity and deception in the virtual community,” in *Communities in cyberspace*. Routledge, 2002, pp. 37–68.
- [81] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.
- [82] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, “Palm-e: An embodied multimodal language model,” *arXiv preprint arXiv:2303.03378*, 2023.
- [83] C. Duffy and C. Thorbecke, “Elon musk said twitter has seen a ‘massive drop in revenue’ as more brands pause ads,” <https://www.edition.cnn.com/2022/11/04/tech/twitter-advertisers/index.html>, 2022, accessed: 2022-12-01.
- [84] C. Duncombe, “Social media and the visibility of horrific violence,” *International Affairs*, vol. 96, no. 3, pp. 609–629, 2020.
- [85] F. Eight and E. King, “Twitter user gender classification,” <https://www.kaggle.com/datasets/crowdflower/twitter-user-gender-classification>, 2016, accessed: 2023-06-01.
- [86] A. ElAzab, “Fake accounts detection in twitter based on minimum weighted feature,” *World*, 2016.
- [87] “2022 strengthened code of practice on disinformation,” <https://digital-strategy.ec.europa.eu/en/library/2022-strengthened-code-practice-disinformation>, European Commission, 2022, accessed: 2022-12-01.
- [88] “The digital services act package,” <https://digital-strategy.ec.europa.eu/en/policies/digital-services-act-package>, European Commission, 2022, accessed: 2022-12-01.

- [89] European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council. [Online]. Available: <https://data.europa.eu/eli/reg/2016/679/oj>
- [90] External Data Source, “dnstwist,” 2018. [Online]. Available: https://www.impactcybertrust.org/dataset_view?idDataset=1318
- [91] M. Fabien, E. Villatoro-Tello, P. Motlicek, and S. Parida, “Bertaa: Bert fine-tuning for authorship attribution,” in *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, 2020, pp. 127–137.
- [92] H. Face, “Marvin,” <https://huggingface.co/learn/audio-course/en/chapter7/voice-assistant#marvin->, 2024, accessed: 2024-06-15.
- [93] Facebook, “An explanation of captchas,” http://www.facebook.com/note.php?note_id=36280205765, 2008, Accessed: 2023-10-15.
- [94] —, “The facebook blog: A continuous commitment to security,” <https://www.facebook.com/blog/blog.php?post=486790652130>, 2011, Accessed: 2023-10-15.
- [95] D. Fernandes *et al.*, “Generalized differential privacy for text obfuscation,” in *Proceedings of the 2018 Conference on Privacy*, 2018.
- [96] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, “The rise of social bots,” *Communications of the ACM*, vol. 59, no. 7, pp. 96–104, 2016.
- [97] —, “The rise of social bots,” *Communications of the ACM*, vol. 59, no. 7, pp. 96–104, 2016.
- [98] M. Feurer and F. Hutter, *Hyperparameter Optimization*. Cham: Springer International Publishing, 2019, pp. 3–33. [Online]. Available: https://doi.org/10.1007/978-3-030-05318-5_1

- [99] J. Fisher, X. Lu, J. Jung, L. Jiang, Z. Harchaoui, and Y. Choi, “Jamdec: Un-supervised authorship obfuscation using constrained decoding over small language models,” *arXiv preprint arXiv:2402.08192*, 2024.
- [100] A. Friggeri, L. Adamic, D. Eckles, and J. Cheng, “Rumor cascades,” *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 8, no. 1, pp. 101–110, May 2014. [Online]. Available: <https://ojs.aaai.org/index.php/ICWSM/article/view/14559>
- [101] Q. Fu, B. Feng, D. Guo, and Q. Li, “Combating the evolving spammers in online social networks,” *Computers & Security*, vol. 72, 09 2017.
- [102] D. Fung, “Drawing for democracy,” <https://www.fairplanet.org/story/political-cartoonist-in-hong-kong/>, 2024, accessed: 2024-07-26.
- [103] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. March, and V. Lempitsky, “Domain-adversarial training of neural networks,” *Journal of machine learning research*, vol. 17, no. 59, pp. 1–35, 2016.
- [104] A. Gatlin, “Eli lilly dives after fake twitter account promises free insulin; takes novo nordisk, sanofi with it,” <https://www.investors.com/news/technology/lly-stock-dives-taking-novo-sanofi-with-it-after-fake-twitter-account-promises-free-insulin/>, 2022, accessed: 2022-12-01.
- [105] T. GILLESPIE, *Custodians of the Internet: Platforms, Content Moderation, and the Hidden Decisions that Shape Social Media*. Yale University Press, 2018.
- [106] T. Gillespie, “Content moderation, ai, and the question of scale,” *Big Data & Society*, vol. 7, no. 2, p. 2053951720943234, 2020.
- [107] P. A. Gimenes, N. T. Roman, and A. M. Carvalho, “Spelling Error Patterns in Brazilian Portuguese,” *Computational Linguistics*, vol. 41, no. 1, pp. 175–183, 03 2015. [Online]. Available: https://doi.org/10.1162/COLI_a_00216

- [108] O. Goga, G. Venkatadri, and K. P. Gummadi, “The doppelgänger bot attack: Exploring identity impersonation in online social networks,” in *Proceedings of the 2015 internet measurement conference*, 2015, pp. 141–153.
- [109] I. Goodfellow, Y. Bengio, and A. Courville, “Softmax units for multinoulli output distributions. deep learning,” 2018.
- [110] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” *Advances in Neural Information Processing Systems*, vol. 3, 06 2014.
- [111] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” *arXiv preprint arXiv:1412.6572*, 2014.
- [112] N. Graham, G. Hirst, and B. Marthi, “Segmenting documents by stylistic character,” *Natural Language Engineering*, vol. 11, no. 4, pp. 397–415, 2005.
- [113] C. Grier, K. Thomas, V. Paxson, and C. M. Zhang, “@spam: the underground on 140 characters or less,” in *CCS ’10*, 2010.
- [114] J. Grieve, “Quantitative authorship attribution: An evaluation of techniques,” *Literary and linguistic computing*, vol. 22, no. 3, pp. 251–270, 2007.
- [115] T. Gröndahl and N. Asokan, “Effective writing style imitation via combinatorial paraphrasing,” *arXiv preprint arXiv:1905.13464*, 2019.
- [116] E. Grossi and M. Buscema, “Introduction to artificial neural networks,” *European journal of gastroenterology & hepatology*, vol. 19, pp. 1046–54, 01 2008.
- [117] N. Guimaraes, A. Figueira, and L. Torgo, “Knowledge-based reliability metrics for social media accounts,” in *WEBIST*, 11 2020.
- [118] L. Guo, J. Liu, X. Zhu, P. Yao, S. Lu, and H. Lu, “Normalized and geometry-aware self-attention network for image captioning,” *2020 IEEE/CVF Conference on*

- Computer Vision and Pattern Recognition (CVPR)*, pp. 10 324–10 333, 2020.
[Online]. Available: <https://api.semanticscholar.org/CorpusID:213176464>
- [119] Y. Guo, C. Li, C. Roan, S. Pakhomov, and T. Cohen, “Crossing the “cookie theft” corpus chasm: applying what bert learns from outside data to the adress challenge dementia detection task,” *Frontiers in Computer Science*, vol. 3, p. 642517, 2021.
- [120] S. Gurses, R. Rizk, and O. Gunther, “Privacy design in online social networks: Learning from privacy breaches and community feedback,” 2008.
- [121] “Twitter - bug boundy program,” <https://hackerone.com/twitter?type=team>, HackerOne, accessed: 2023-03-25.
- [122] Y. Hao, Z. Chi, L. Dong, and F. Wei, “Optimizing prompts for text-to-image generation,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [123] D. Harwell, “A fake tweet sparked panic at eli lilly and may have cost twitter millions,” <https://www.washingtonpost.com/technology/2022/11/14/twitter-fake-eli-lilly/>, 2022, accessed: 2022-12-01.
- [124] R. Haulcy and J. Glass, “Classifying alzheimer’s disease using audio and text-based representations of speech,” *Frontiers in Psychology*, vol. 11, p. 624137, 2021.
- [125] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [126] —, “Identity mappings in deep residual networks,” in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*. Springer, 2016, pp. 630–645.
- [127] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, *Image Captioning: Transforming Objects into Words*. Red Hook, NY, USA: Curran Associates Inc., 2019.

- [128] G. Hirst and O. Feiguina, “Bigrams of syntactic labels for authorship discrimination of short texts,” *Literary and Linguistic Computing*, vol. 22, no. 4, pp. 405–417, 2007.
- [129] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS ’20. Red Hook, NY, USA: Curran Associates Inc., 2020.
- [130] T. K. Ho, “Random decision forests,” in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.
- [131] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, pp. 1735–80, 12 1997.
- [132] D. I. Holmes, “The evolution of stylometry in humanities scholarship,” *Literary and linguistic computing*, vol. 13, no. 3, pp. 111–117, 1998.
- [133] D. I. Holmes and J. Kardos, “Who was the author? an introduction to stylometry,” *Chance*, vol. 16, no. 2, pp. 5–8, 2003.
- [134] K. Holt, “How to spot a fake verified twitter account,” <https://www.dailydot.com/unclick/how-to-spot-fake-verified-twitter/>, 2013, accessed: 2023-04-01.
- [135] D. Holz, “Midjourney,” <https://www.midjourney.com/home>, accessed: 2024-05-10.
- [136] D. L. Hoover, “Another perspective on vocabulary richness,” *Computers and the Humanities*, vol. 37, pp. 151–178, 2003.
- [137] E. Howe, “Information commissioner’s office,” <https://ico.org.uk/>, accessed: 2023-12-01.
- [138] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, “Squeeze-and-excitation networks,” 2019.
- [139] Y. Hu, H. Wang, R. He, L. Li, G. Tyson, I. Castro, Y. Guo, L. Wu, and G. Xu, “Mobile app squatting,” in *Proceedings of The Web Conference 2020*, 04 2020, pp. 1727–1738.

- [140] Z. Hu, R. K.-W. Lee, L. Wang, E.-p. Lim, and B. Dai, “Deepstyle: User style embedding for authorship attribution of short texts,” in *Web and Big Data: 4th International Joint Conference, APWeb-WAIM 2020, Tianjin, China, September 18-20, 2020, Proceedings, Part II 4*. Springer, 2020, pp. 221–229.
- [141] B. Huang, C. Chen, and K. Shu, “Can large language models identify authorship?” *arXiv preprint arXiv:2403.08213*, 2024.
- [142] S. Huang, L. Dong, W. Wang, Y. Hao, S. Singhal, S. Ma, T. Lv, L. Cui, O. K. Mohammed, B. Patra *et al.*, “Language is not all you need: Aligning perception with language models,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [143] P.-M. Hui, K.-C. Yang, C. Torres-Lugo, Z. Monroe, M. McCarty, B. Serrette, V. Pentchev, and F. Menczer, “Botslayer: real-time detection of bot amplification on twitter,” *Journal of Open Source Software*, vol. 4, p. 1706, 10 2019.
- [144] Instagram, “Impersonation accounts,” <https://help.instagram.com/446663175382270>, accessed: 2021-10-01.
- [145] —, “Instagram,” <https://www.instagram.com/>, accessed: 2021-10-01.
- [146] ItalianPostNews, “Twitter, from apple to tesla the fake tweets with the “blue check” that have become memes,” <https://www.italianpost.news/twitter-from-apple-to-tesla-the-fake-tweets-with-the-blue-check-that-have-become-memes/>, 2022, accessed: 2022-12-01.
- [147] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, “Social phishing,” *Communications of the ACM*, vol. 50, no. 10, pp. 94–100, 2007.
- [148] R. Jiang, G.-C. Zheng, T. Li, T.-R. Yang, J.-D. Wang, and X. Li, “A survey of multi-modal controllable diffusion models,” *Journal of Computer Science and Technology*, vol. 39, no. 3, pp. 509–541, 2024.

- [149] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, “Is bert really robust? a strong baseline for natural language attack on text classification and entailment,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 05, 2020, pp. 8018–8025.
- [150] L. Jin, D. Takabi, and J. Joshi, “Towards active detection of identity clone attacks on online social networks,” in *Proceedings of the first ACM conference on Data and application security and privacy*, 02 2011, pp. 27–38.
- [151] G. Kacmarcik and M. Gamon, “Obfuscating document stylometry to preserve author anonymity,” in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, 2006, pp. 444–451.
- [152] B. Kahle and B. Gilliat, “Internet archive: Wayback machine,” <https://archive.org/web/>, accessed: 2021-12-5.
- [153] I. Kandel and M. Castelli, “Transfer learning with convolutional neural networks for diabetic retinopathy image classification. a review,” *Applied Sciences*, vol. 10, p. 2021, 03 2020.
- [154] A. M. Kaplan and M. Haenlein, “Users of the world, unite! the challenges and opportunities of social media,” *Business horizons*, vol. 53, no. 1, pp. 59–68, 2010.
- [155] S. Karen, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv: 1409.1556*, 2014.
- [156] R. Kasakowskij, N. Friedrich, K. J. Fietkiewicz, and W. G. Stock, “Anonymous and non-anonymous user behavior on social media: A case study of jodel and instagram,” *Journal of Information Science Theory and Practice*, vol. 6, no. 3, pp. 25–36, 2018.
- [157] Kaspersky, “Scammers with blue checkmarks on x.” [Online]. Available: <https://www.kaspersky.com/blog/beware-of-twitter-blue-fake-accounts/49199/>
- [158] Keras, “Imagedatagenerator,” https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator, 2024, accessed: 2024-02-01.

- [159] M. Kestemont, “Function words in authorship attribution. from black magic to theory?” in *Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLFL)*, 2014, pp. 59–66.
- [160] N. Ketkar, *Stochastic Gradient Descent*. Springer, 10 2017, pp. 111–130.
- [161] Y. Kim, “Convolutional neural networks for sentence classification,” in *Conference on Empirical Methods in Natural Language Processing*, 2014. [Online]. Available: <https://arxiv.org/abs/1408.5882>
- [162] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations*, 12 2014.
- [163] P. Kintis, N. Miramirkhani, C. Lever, Y. Chen, R. Romero-Gómez, N. Pitropakis, N. Nikiforakis, and M. Antonakakis, “Hiding in plain sight: A longitudinal study of combosquatting abuse,” in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 08 2017.
- [164] K. Klonick, “The new governors: The people, rules, and processes governing online speech. harvard law review 131,” 2018.
- [165] J. A. Konstan and J. Riedl, “Recommender systems: from algorithms to user experience,” *User modeling and user-adapted interaction*, vol. 22, pp. 101–123, 2012.
- [166] G. Kontaxis, I. Polakis, S. Ioannidis, and E. Markatos, “Detecting social network profile cloning,” in *2011 IEEE international conference on pervasive computing and communications workshops (PERCOM Workshops)*, 04 2011, pp. 295 – 300.
- [167] R. Koo, M. Lee, V. Raheja, J. I. Park, Z. M. Kim, and D. Kang, “Benchmarking cognitive biases in large language models as evaluators,” *arXiv preprint arXiv:2309.17012*, 2023.
- [168] R. M. Kowalski, G. W. Giumetti, A. N. Schroeder, and M. R. Lattanner, “Bullying in the digital age: a critical review and meta-analysis of cyberbullying research among youth.” *Psychological bulletin*, vol. 140, no. 4, p. 1073, 2014.

- [169] J. Krumm, “Inference attacks on location tracks,” in *Pervasive Computing: 5th International Conference, PERVASIVE 2007, Toronto, Canada, May 13-16, 2007. Proceedings 5*. Springer, 2007, pp. 127–143.
- [170] D. Kumar, R. Paccagnella, P. Murley, E. Hennenfent, J. Mason, A. Bates, and M. Bailey, “Skill squatting attacks on amazon alexa,” in *27th USENIX Security Symposium (USENIX Security 18)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 33–47. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity18/presentation/kumar>
- [171] S. Kumar, J. Cheng, J. Leskovec, and V. Subrahmanian, “An army of me: Sock-puppets in online discussion communities,” in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 857–866.
- [172] S. Kumar and N. Shah, “False information on web and social media: A survey,” *arXiv preprint arXiv:1804.08559*, 2018.
- [173] A. Lab, “Squad,” <https://github.com/APSS-Imperial/SQUAD>, 2023.
- [174] —, “Squad,” <https://github.com/APSS-Imperial/SQUAD/blob/main/LICENSE>, 2023.
- [175] —, “Squad framework,” <https://sites.google.com/view/squad-framework/home>, 2023.
- [176] K. Lagutina, N. Lagutina, E. Boychuk, I. Vorontsova, E. Shliakhtina, O. Belyaeva, I. Paramonov, and P. Demidov, “A survey on stylometric text features,” in *2019 25th Conference of Open Innovations Association (FRUCT)*. IEEE, 2019, pp. 184–195.
- [177] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “Albert: A lite bert for self-supervised learning of language representations,” *arXiv preprint arXiv:1909.11942*, 2019.
- [178] K. Lee, J. Caverlee, and S. Webb, “Uncovering social spammers: Social honeypots + machine learning,” in *Proceedings of the 33rd international ACM SIGIR*

- conference on Research and development in information retrieval*, ser. SIGIR '10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 435–442. [Online]. Available: <https://doi.org/10.1145/1835449.1835522>
- [179] S. Lee and J. Kim, “Warningbird: A near real-time detection system for suspicious urls in twitter stream,” *Dependable and Secure Computing, IEEE Transactions on*, vol. 10, pp. 183–195, 05 2013.
- [180] A. Lepipas, A. Borovykh, and S. Demetriou, “Username squatting on online social networks: A study on x,” in *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, ser. ASIA CCS '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 621–637. [Online]. Available: <https://doi.org/10.1145/3634737.3637637>
- [181] B. Lester, R. Al-Rfou, and N. Constant, “The power of scale for parameter-efficient prompt tuning,” in *Conference on Empirical Methods in Natural Language Processing*, 2021. [Online]. Available: <https://api.semanticscholar.org/CorpusID:233296808>
- [182] Let’s Encrypt, “Let’s encrypt - free ssl/tls certificates,” <https://letsencrypt.org>, 2017, accessed: 2023-06-01.
- [183] V. Levenshtein, “Wikipedia levenshtein distance,” https://en.wikipedia.org/wiki/Levenshtein_distance, accessed: 2021-10-18.
- [184] Lexica, “Lexica art,” <https://lexica.art/>, 2022, accessed: 2024-02-01.
- [185] G. Li, L. Zhu, P. Liu, and Y. Yang, “Entangled transformer for image captioning,” *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8927–8936, 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:207982683>

- [186] H. Li, S. J. Pan, S. Wang, and A. C. Kot, “Domain generalization with adversarial feature learning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5400–5409.
- [187] J. Li, G. Longinos, S. Wilson, and W. Magdy, “Emoji and self-identity in twitter bios,” in *Proceedings of the fourth workshop on natural language processing and computational social science*, 2020, pp. 199–211.
- [188] J. Li, D. Li, S. Savarese, and S. Hoi, “Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models,” in *International conference on machine learning*. PMLR, 2023, pp. 19 730–19 742.
- [189] J. Li, D. Li, C. Xiong, and S. Hoi, “Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation,” in *International conference on machine learning*. PMLR, 2022, pp. 12 888–12 900.
- [190] X. L. Li and P. Liang, “Prefix-tuning: Optimizing continuous prompts for generation,” *arXiv preprint arXiv:2101.00190*, 2021.
- [191] C.-Y. Lin, “Rouge: A package for automatic evaluation of summaries,” in *Text summarization branches out*, 2004, pp. 74–81.
- [192] F. Liu and Y. Liu, “Correlation between rouge and human evaluation of extractive meeting summaries,” in *Proceedings of ACL-08: HLT, short papers*, 2008, pp. 201–204.
- [193] H. Liu, C. Li, Q. Wu, and Y. J. Lee, “Visual instruction tuning,” in *NeurIPS*, 2023.
- [194] X. Liu, K. Ji, Y. Fu, W. L. Tam, Z. Du, Z. Yang, and J. Tang, “P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks,” *arXiv preprint arXiv:2110.07602*, 2021.
- [195] —, “P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks,” *arXiv preprint arXiv:2110.07602*, 2021.

- [196] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.
- [197] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.
- [198] S. Loria, “textblob documentation,” *Release 0.15*, vol. 2, 2018.
- [199] D. G. Lowe, “Object recognition from local scale-invariant features,” in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.
- [200] Y. LU, “Kaggle fake account dataset,” <https://www.kaggle.com/bitandatom/social-network-fake-account-dataset>, accessed: 2021-10-14.
- [201] Y. Luo, J. Jiayi, X. Sun, L. Cao, W. Yongjian, F. Huang, C.-W. Lin, and R. Ji, “Dual-level collaborative transformer for image captioning,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 2286–2293, 05 2021.
- [202] S. Luz, F. Haider, S. de la Fuente, D. Fromm, and B. MacWhinney, “Alzheimer’s dementia recognition through spontaneous speech: The adress challenge,” *arXiv preprint arXiv:2004.06833*, 2020.
- [203] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, “Hate speech detection: Challenges and solutions,” *PloS one*, vol. 14, no. 8, p. e0221152, 2019.
- [204] D. Macko, R. Moro, A. Uchendu, J. S. Lucas, M. Yamashita, M. Pikuliak, I. Srba, T. Le, D. Lee, J. Simko *et al.*, “Multitude: Large-scale multilingual machine-generated text detection benchmark,” *arXiv preprint arXiv:2310.13606*, 2023.

- [205] D. Macko, R. Moro, A. Uchendu, I. Srba, J. S. Lucas, M. Yamashita, N. I. Tripto, D. Lee, J. Simko, and M. Bielikova, “Authorship obfuscation in multilingual machine-generated text detection,” *arXiv preprint arXiv:2401.07867*, 2024.
- [206] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” *ArXiv*, vol. abs/1706.06083, 2017. [Online]. Available: <https://api.semanticscholar.org/CorpusID:3488815>
- [207] P. Mahajan and V. Baths, “Acoustic and language based deep learning approaches for alzheimer’s dementia detection from spontaneous speech. front aging neurosci. 2021; 13,” 2021.
- [208] A. Mahmood, F. Ahmad, Z. Shafiq, P. Srinivasan, and F. Zaffar, “A girl has no name: Automated authorship obfuscation using mutant-x,” *Proceedings on Privacy Enhancing Technologies*, 2019.
- [209] M. Majka, *naivebayes: High Performance Implementation of the Naive Bayes Algorithm in R*, 2019, r package version 0.9.7. [Online]. Available: <https://CRAN.R-project.org/package=naivebayes>
- [210] M. B. Malyutov, “Authorship attribution of texts: A review,” *General Theory of Information Transfer and Combinatorics*, pp. 362–380, 2006.
- [211] F. S. Marcondes, J. J. Almeida, and P. Novais, “Structural onomatologic for username generation: A partial account,” in *STAIRS@ECAI*, 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:221839940>
- [212] E. Mariconti, J. Onaolapo, S. Ahmad, N. Nikiforou, M. Egele, N. Nikiforakis, and G. Stringhini, “Why allowing profile name reuse is a bad idea,” in *Proceedings of the 9th European Workshop on System Security*, 04 2016, pp. 1–6.
- [213] E. Mariconti, J. Onaolapo, S. S. Ahmad, N. Nikiforou, M. Egele, N. Nikiforakis, and G. Stringhini, “What’s in a name? understanding profile name reuse on twitter,”

- in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 1161–1170.
- [214] M. Martinc, F. Haider, S. Pollak, and S. Luz, “Temporal integration of text transcripts and acoustic features for alzheimer’s diagnosis based on spontaneous speech,” *Frontiers in Aging Neuroscience*, vol. 13, p. 642647, 2021.
- [215] J. N. Matias, “Preventing harassment and increasing group participation through social norms in 2,190 online science discussions,” *Proceedings of the National Academy of Sciences*, vol. 116, no. 20, pp. 9785–9789, 2019.
- [216] J. Mattern, B. Weggenmann, and F. Kerschbaum, “The limits of word level differential privacy,” *arXiv preprint arXiv:2205.02130*, 2022.
- [217] M. McHugh, “Interrater reliability: The kappa statistic,” *Biochemia medica : časopis Hrvatskoga društva medicinskih biokemičara / HDMB*, vol. 22, pp. 276–82, 10 2012.
- [218] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, “Exploiting unintended feature leakage in collaborative learning,” in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 691–706.
- [219] —, “Exploiting unintended feature leakage in collaborative learning,” in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 691–706.
- [220] Meta, “Facebook,” <https://www.facebook.com/>, accessed: 2021-10-01.
- [221] —, “Facebook community standards,” <https://www.facebook.com/communitystandards/>, accessed: 2021-10-01.
- [222] Midjourney, “Midjourney subreddit.” <https://www.reddit.com/r/midjourney/>, accessed: 2024-07-11.

- [223] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *Advances in neural information processing systems*, vol. 26, 2013.
- [224] G. K. Mikros and E. K. Argiri, “Investigating topic influence in authorship attribution.” in *PAN*, 2007.
- [225] G. A. Miller, “Wordnet: a lexical database for english,” *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [226] T. Millington and S. Luz, “Analysis and classification of word co-occurrence networks from alzheimer’s patients and controls,” *Frontiers in Computer Science*, vol. 3, p. 649508, 2021.
- [227] A. Mohaisen and J. Kim, “The sybil attacks and defenses: A survey,” *Smart Comput. Rev.*, vol. 3, pp. 480–489, 2013.
- [228] A. Mucherino, P. J. Papajorgji, and P. M. Pardalos, *k-Nearest Neighbor Classification*. New York, NY: Springer New York, 2009, pp. 83–106. [Online]. Available: https://doi.org/10.1007/978-0-387-88615-2_4
- [229] L. Nguyen, Y. Tian, S. Cho, W. Kwak, S. Parab, Y. Kim, P. Tague, and J. Zhang, “Unlocin: Unauthorized location inference on smartphones without being caught,” in *2013 International Conference on Privacy and Security in Mobile Systems (PRISMS)*. IEEE, 2013, pp. 1–8.
- [230] T. D. Nguyen, A. Azizpour, and M. C. Stamm, “Forensic self-descriptions are all you need for zero-shot detection, open-set source attribution, and clustering of ai-generated images,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 3040–3050.
- [231] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, “Glide: Towards photorealistic image generation and editing with text-guided diffusion models,” 12 2021.

- [232] G. Nie, C. Kim, Y. Yang, and Y. Ren, “Attributing image generative models using latent fingerprints,” in *International Conference on Machine Learning*. PMLR, 2023, pp. 26 150–26 165.
- [233] N. Nikiforakis, M. Balduzzi, L. Desmet, F. Piessens, and W. Joosen, “Soundsquatting: Uncovering the use of homophones in domain squatting,” in *Information Security: 17th International Conference, ISC 2014, Hong Kong, China, October 12-14, 2014. Proceedings 17*, 10 2014, pp. 291–308.
- [234] P. Norvig, “English letter frequency counts: Mayzner revisited or *etaoin srhldcu*,” <http://norvig.com/mayzner.html>, 2013, accessed 5 June 2025.
- [235] J. Oak, “Language models for authorship obfuscation,” *arXiv preprint arXiv:2205.02130*, 2022.
- [236] Oberlo, “Twitter statistics,” <https://www.oberlo.com/blog/twitter-statistics>, 2023, accessed: 2023-11-01.
- [237] R. Oentaryo, A. Murdopo, P. K. Prasetyo, and E.-P. Lim, “On profiling bots in social media,” in *Social Informatics: 8th International Conference, SocInfo 2016, Bellevue, WA, USA, November 11-14, 2016, Proceedings, Part I 8*, 09 2016.
- [238] A. Olteanu, C. Castillo, J. Boy, and K. Varshney, “The effect of extremist violence on hateful speech online,” in *Proceedings of the international AAAI conference on web and social media*, vol. 12, no. 1, 2018.
- [239] OpenAI, “Dall·e 3 system card,” 2023, version 2.
- [240] —, “Gpt-4v(ision) system card,” 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:263218031>
- [241] —, “Clip large,” <https://huggingface.co/openai/clip-vit-large-patch14>, 2024, accessed: 2024-02-01.
- [242] OpenAI, “Openai,” <https://openai.com>, 2024, accessed: 2024-02-01.

- [243] P. PANDEY, “Kaggle popular accounts dataset,” <https://www.kaggle.com/parulpandey/100-mostfollowed-twitter-accounts-as-of-dec2019>, accessed: 2021-09-30.
- [244] O. Papakyriakopoulos, J. C. M. Serrano, and S. Hegelich, “The spread of covid-19 conspiracy theories on social media and the effect of content moderation,” *The Harvard Kennedy School (HKS) Misinformation Review*, vol. 18, 2020.
- [245] A. Pathak, N. Madani, and K. Joseph, “A method to analyze multiple social identities in twitter bios,” *Proceedings of the ACM on human-computer interaction*, vol. 5, no. CSCW2, pp. 1–35, 2021.
- [246] Patreon, “Voxta,” <https://www.patreon.com/Voxta>, 2024, accessed: 2024-06-15.
- [247] E. Pearson, “All the world wide web’s a stage: The performance of identity in online social networks,” *First Monday*, 2009.
- [248] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [249] X. Peng, B. Usman, K. Saito, N. Kaushik, J. Hoffman, and K. Saenko, “Syn2real: A new benchmark for synthetic-to-real visual domain adaptation,” *arXiv preprint arXiv:1806.09755*, 2018.
- [250] J. Pennington, R. Socher, and C. Manning, “GloVe: Global vectors for word representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. [Online]. Available: <https://aclanthology.org/D14-1162>

- [251] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
- [252] Z. Pesochinsky, “Almost famous: Preventing username-squatting on social networking websites,” *Cardozo Arts & Ent. LJ*, vol. 28, p. 223, 2010.
- [253] Y. Qu, X. Shen, X. He, M. Backes, S. Zannettou, and Y. Zhang, “Unsafe diffusion: On the generation of unsafe images and hateful memes from text-to-image models,” in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023, pp. 3403–3417.
- [254] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
- [255] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020.
- [256] M. A. Ramadhan, D. P. Sari, and A. Musdholifah, “Author obfuscation in indonesian news articles using genetic algorithms,” in *Proceedings of the 2021 International Conference on Computational Linguistics*, 2021.
- [257] D. Ramalingam and V. Chinnaiyah, “Fake profile detection techniques in large-scale online social networks: A comprehensive review,” *Computers & Electrical Engineering*, vol. 65, pp. 165–177, 2018.
- [258] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, “Hierarchical text-conditional image generation with clip latents,” 04 2022.
- [259] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, “Zero-shot text-to-image generation,” 02 2021.

- [260] A. Rauchfleisch and J. Kaiser, “The false positive problem of automatic bot detection in social science research,” *PLoS ONE*, vol. 15, 10 2020.
- [261] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative adversarial text to image synthesis,” in *International conference on machine learning*. PMLR, 2016, pp. 1060–1069.
- [262] N. Reimann and C. Porterfield, “Musk says apple cutting twitter ads—here are the other companies rethinking their ties,” <https://www.forbes.com/sites/nicholasreimann/2022/11/28/musk-says-apple-cutting-twitter-ads-here-are-the-other-companies-rethinking-their-ties/?sh=5efc41b77032>, 2022, accessed: 2022-12-01.
- [263] N. Reimers, “Sentence-bert: Sentence embeddings using siamese bert-networks,” *arXiv preprint arXiv:1908.10084*, 2019.
- [264] N. Reimers and I. Gurevych, “Sentence transformers,” <https://www.huggingface.co/sentence-transformers/all-mpnet-base-v2>, 2024, accessed: 2024-03-01.
- [265] M. Rigaki and S. Garcia, “A survey of privacy attacks in machine learning,” *ACM Computing Surveys*, vol. 56, no. 4, pp. 1–34, 2023.
- [266] E. Riloff, S. Patwardhan, and J. Wiebe, “Feature subsumption for opinion analysis,” in *Proceedings of the 2006 conference on empirical methods in natural language processing*, 2006, pp. 440–448.
- [267] S. T. Roberts, *Behind the screen*. Yale University Press, 2019.
- [268] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 674–10 685, 06 2022.
- [269] —, “High-resolution image synthesis with latent diffusion models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

- [270] A. Roy, A. Shah, K. Shah, A. Roy, and R. Chellappa, “Cap2aug: Caption guided image to image data augmentation,” *arXiv preprint arXiv:2212.05404*, 2022.
- [271] S. Ruder, P. Ghaffari, and J. G. Breslin, “Character-level and multi-channel convolutional neural networks for large-scale authorship attribution,” *arXiv preprint arXiv:1609.06686*, 2016.
- [272] —, “Character-level and multi-channel convolutional neural networks for large-scale authorship attribution,” *arXiv preprint arXiv:1609.06686*, 2016.
- [273] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, pp. 211–252, 2015.
- [274] —, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, pp. 211–252, 2015.
- [275] A. Sablayrolles, M. Douze, C. Schmid, Y. Ollivier, and H. Jégou, “White-box vs black-box: Bayes optimal strategies for membership inference,” in *International Conference on Machine Learning*. PMLR, 2019, pp. 5558–5567.
- [276] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, “Photorealistic text-to-image diffusion models with deep language understanding,” *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.
- [277] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, “ML-leaks: Model and data independent membership inference attacks and defenses on machine learning models,” *arXiv preprint arXiv:1806.01246*, 2018.
- [278] Y. Sari, M. Stevenson, and A. Vlachos, “Topic or style? exploring the most useful features for authorship attribution,” in *Proceedings of the 27th international conference on computational linguistics*, 2018, pp. 343–353.

- [279] Y. Sari, A. Vlachos, and M. Stevenson, “Continuous n-gram representations for authorship attribution,” in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017, pp. 267–273.
- [280] G. Sartor and F. Lagioia, “The impact of the general data protection regulation (gdpr) on artificial intelligence,” 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:229631526>
- [281] M. Sayyadiharikandeh, O. Varol, K.-C. Yang, A. Flammini, and F. Menczer, “Detection of novel social bots by ensembles of specialized classifiers,” *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 10 2020. [Online]. Available: <http://dx.doi.org/10.1145/3340531.3412698>
- [282] A. Schmidt and M. Wiegand, “A survey on hate speech detection using natural language processing,” in *Proceedings of the fifth international workshop on natural language processing for social media*, 2017, pp. 1–10.
- [283] P. Schramowski, M. Brack, B. Deiseroth, and K. Kersting, “Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 522–22 531.
- [284] C. Schuhmann, R. Vencu, R. Beaumont, R. Kaczmarczyk, C. Mullis, A. Katta, T. Coombes, J. Jitsev, and A. Komatsuzaki, “Laion-400m: Open dataset of clip-filtered 400 million image-text pairs,” *arXiv preprint arXiv:2111.02114*, 2021.
- [285] —, “Laion-400m: Open dataset of clip-filtered 400 million image-text pairs,” *arXiv preprint arXiv:2111.02114*, 2021.
- [286] R. Schwartz, O. Tsur, A. Rappoport, and M. Koppel, “Authorship attribution of micro-messages,” in *Proceedings of the 2013 Conference on empirical methods in natural language processing*, 2013, pp. 1880–1891.

- [287] S. Seelan, K. Kavin, and A. Hassan, “Frustrate twitter from automation: How far a user can be trusted?” in *2013 International Conference on Human Computer Interactions (ICHCI)*, 08 2013, pp. 1–5.
- [288] K. Semertzidis, E. Pitoura, and P. Tsaparas, “How people describe themselves on twitter,” in *Proceedings of the ACM SIGMOD Workshop on Databases and Social Networks*, 2013, pp. 25–30.
- [289] Y. Seroussi, I. Zukerman, and F. Bohnert, “Authorship attribution with topic models,” *Computational Linguistics*, vol. 40, no. 2, pp. 269–310, 2014.
- [290] Z. Sha, Z. Li, N. Yu, and Y. Zhang, “De-fake: Detection and attribution of fake images generated by text-to-image generation models,” in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 11 2023, pp. 3418–3432.
- [291] V. Shankar, R. Roelofs, H. Mania, A. Fang, B. Recht, and L. Schmidt, “Evaluating machine accuracy on imagenet,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 8634–8644.
- [292] C. Shao, G. Ciampaglia, O. Varol, A. Flammini, and F. Menczer, “The spread of fake news by social bots,” *arXiv preprint arXiv:1707.07592*, 07 2017.
- [293] R. Shetty, B. Schiele, and M. Fritz, “{A4NT}: Author attribute anonymity by adversarial training of neural machine translation,” in *27th USENIX Security Symposium (USENIX Security 18)*, 2018, pp. 1633–1650.
- [294] —, “{A4NT}: Author attribute anonymity by adversarial training of neural machine translation,” in *27th USENIX Security Symposium (USENIX Security 18)*, 2018, pp. 1633–1650.
- [295] N. Shoeibi, N. Shoeibi, P. Chamoso, Z. Alizadehsani, and J. M. Corchado, “A hybrid model for the measurement of the similarity between twitter profiles,” *Sustainability*, vol. 14, no. 9, p. 4909, 2022.

- [296] M. Shokri, S. I. Levitan, and R. Levitan, “Personalized author obfuscation with large language models,” *arXiv preprint arXiv:2505.12090*, 2025.
- [297] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *2017 IEEE symposium on security and privacy (SP)*. IEEE, 2017, pp. 3–18.
- [298] P. Shrestha, S. Sierra, F. A. González, M. Montes, P. Rosso, and T. Solorio, “Convolutional neural networks for authorship attribution of short texts,” in *Proceedings of the 15th conference of the European chapter of the association for computational linguistics: Volume 2, short papers*, 2017, pp. 669–674.
- [299] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, “Fake news detection on social media: A data mining perspective,” *ACM SIGKDD Explorations Newsletter*, vol. 19, 08 2017.
- [300] L. Shu, L. Luo, J. Hoskerc, Y. Zhu, Y. Liu, S. Tong, J. Chen, and L. Meng, “Rewritelm: An instruction-tuned large language model for text rewriting,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 17, 2024, pp. 18 970–18 980.
- [301] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv 1409.1556*, 09 2014.
- [302] C. Song and A. Raghunathan, “Information leakage in embedding models,” in *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, 2020, pp. 377–390.
- [303] C. Song and V. Shmatikov, “Overlearning reveals sensitive attributes,” 2020. [Online]. Available: <https://arxiv.org/abs/1905.11742>
- [304] G. Sood, *virustotal: R Client for the virustotal API*, 2021, r package version 0.2.2.

- [305] E. Stamatatos, "A survey of modern authorship attribution methods," *Journal of the American Society for information Science and Technology*, vol. 60, no. 3, pp. 538–556, 2009.
- [306] Stiffl, "Reddit usernames from subreddit - scraper," <https://apify.com/stiffl/reddit-username-from-subreddit>, accessed: 2024-07-11.
- [307] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting Spammers on Social Networks," in *Proceedings of the 26th Annual Computer Security Applications Conference*, ser. ACSAC. New York, NY, USA: ACM, 2010, pp. 1–9. [Online]. Available: <https://doi.org/10.1145/1920261.1920263>
- [308] V. S. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, and F. Menczer, "The darpa twitter bot challenge," *Computer*, vol. 49, no. 6, pp. 38–46, 2016.
- [309] Sysomos.com, "Inside twitter: An in-depth look inside the twitter world," <https://www.key4biz.it/files/000270/00027033.pdf>, accessed: 2023-06-01.
- [310] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
- [311] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.
- [312] J. Szurdi and N. Christin, "Email typosquatting," in *Proceedings of the 2017 internet measurement conference*, 11 2017.
- [313] J. Szurdi, B. Kocso, G. Cseh, J. Spring, M. Felegyhazi, and C. Kanich, "The long "taile" of typosquatting domain names," in *23rd USENIX Security Symposium (USENIX Security 14)*, 01 2014.

- [314] S. Tardelli, M. Avvenuti, M. Tesconi, and S. Cresci, “Characterizing social bots spreading financial disinformation,” in *International conference on human-computer interaction*. Springer, 2020, pp. 376–392.
- [315] N. Team, “Nltk names corpus.” <https://www.nltk.org/howto/corpus.html>, accessed: 2024-07-11.
- [316] T. Teubner and S. Camacho, “Facing reciprocity: How photos and avatars promote interaction in micro-communities,” *Group Decision and Negotiation*, vol. 32, no. 2, pp. 435–467, 2023.
- [317] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song, “Design and evaluation of a real-time url spam filtering service,” in *2011 IEEE symposium on security and privacy*, 05 2011, pp. 447–462.
- [318] TikTok, “Tiktok,” <https://www.tiktok.com/>, accessed: 2021-10-01.
- [319] Q. Tran, “Worldwide dementia cases,” <https://alzheimersresearchuk.org/news/worldwide-dementia-cases-to-triple-by-2050-to-over-150-million>, 2022, accessed: 2024-06-018.
- [320] N. I. Tripto, S. Venkatraman, D. Macko, R. Moro, I. Srba, A. Uchendu, T. Le, and D. Lee, “A ship of theseus: Curious cases of paraphrasing in llm-generated texts,” *arXiv preprint arXiv:2311.08374*, 2023.
- [321] S. Turkle, “Life on the screen: identity in the age of the internet simon & schuster,” *New York*, 1995.
- [322] Twitter, “Twitter - country withheld content,” <https://help.twitter.com/en/rules-and-policies/tweet-withheld-by-country>, accessed: 2021-11-21.
- [323] —, “Twitter - rules,” <https://help.twitter.com/en/safety-and-security/report-twitter-impersonation>, accessed: 2023-04-03.

- [324] —, “Twitter bots,” https://blog.twitter.com/en_us/topics/company/2020/bot-or-not, accessed: 2021-10-27.
- [325] —, “About twitter blue,” <https://help.twitter.com/en/using-twitter/twitter-blue>, 2022, accessed: 2022-12-01.
- [326] UK Information Commissioner’s Office, “Guidance on ai and data protection—chapter on profiling and inference,” 2020, retrieved from ico.org.uk.
- [327] Undisclosed, “Personal communication with the manager of user support and the product manager of the core and community management teams in tuenti,” 2011.
- [328] J. Valinsky, “Elon musk rebrands twitter as x,” <https://edition.cnn.com/2023/07/24/tech/twitter-rebrands-x-elon-musk-hnk-intl/index.html>, 2023, accessed: 2023-07-24.
- [329] L. Van der Maaten and G. Hinton, “Visualizing data using t-sne.” *Journal of machine learning research*, vol. 9, no. 11, 2008.
- [330] O. Varol, E. Ferrara, C. Davis, F. Menczer, and A. Flammini, “Online human-bot interactions: Detection, estimation, and characterization,” in *Proceedings of the international AAAI conference on web and social media*, vol. 11, no. 1, 2017, pp. 280–289.
- [331] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.
- [332] M. Veale and L. Edwards, “Clarity, surprises, and further questions in the article 29 working party draft guidance on automated decision-making and profiling,” *Computer Law & Security Review*, vol. 34, no. 2, pp. 398–404, 2018.
- [333] VIKAS, “Kaggle bots dataset,” <https://www.kaggle.com/vikasg/russian-troll-tweets>, 2017, accessed: 2021-10-14.

- [334] S. Vosoughi, D. Roy, and S. Aral, “The spread of true and false news online,” *Science*, vol. 359, pp. 1146–1151, 03 2018.
- [335] S. Wachter and B. Mittelstadt, “A right to reasonable inferences: re-thinking data protection law in the age of big data and ai,” *Colum. Bus. L. Rev.*, p. 494, 2019.
- [336] C. F. Wade Alcorn and M. Orru, “Urlcrazy,” <https://morningstarsecurity.com/research/urlcrazy>, accessed: 2021-10-01.
- [337] A. Walker, “Inside the magical world of ai prompters on reddit,” <https://hyperallergic.com/855052/inside-the-magical-world-of-ai-prompters-on-reddit/>, 2024, accessed: 2024-07-28.
- [338] A. Wang, “Don’t follow me - spam detection in twitter.” in *2010 international conference on security and cryptography (SECRYPT)*, 01 2010, pp. 142–151.
- [339] A. H. Wang, “Don’t follow me: Spam detection in twitter,” in *2010 international conference on security and cryptography (SECRYPT)*. IEEE, 2010, pp. 1–10.
- [340] H. Wang, “Defending against authorship identification attacks,” *arXiv preprint arXiv:2310.01568*, 2023.
- [341] W. Y. Wang, ““liar, liar pants on fire”: A new benchmark dataset for fake news detection,” in *Annual Meeting of the Association for Computational Linguistics*, 2017. [Online]. Available: <https://api.semanticscholar.org/CorpusID:10326133>
- [342] Z. J. Wang, E. Montoya, D. Munechika, H. Yang, B. Hoover, and D. H. Chau, “Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models,” *ArXiv*, vol. abs/2210.14896, 2022. [Online]. Available: <https://api.semanticscholar.org/CorpusID:253116574>
- [343] Z. Wang, T. Le, and D. Lee, “Upton: Preventing authorship leakage from public text release via data poisoning,” *arXiv preprint arXiv:2211.09717*, 2022.

- [344] —, “Authorship obfuscation in multilingual machine-generated text detection,” *arXiv preprint arXiv:2401.07867*, 2024.
- [345] C. Wardle and H. Derakhshan, *Information disorder: Toward an interdisciplinary framework for research and policymaking*. Council of Europe Strasbourg, 2017, vol. 27.
- [346] J. Weatherbed, “Elon musk says twitter will begin manually authenticating blue, grey, and gold accounts as soon as next week,” <https://www.theverge.com/2022/11/25/23477550/twitter-manual-verification-blue-checkmark-gold-grey>, 2022, accessed: 2022-12-01.
- [347] B. Weggenmann and F. Kerschbaum, “Syntf: Synthetic and differentially private term frequency vectors for privacy-preserving text mining,” in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 305–314.
- [348] Wikipedia, “Wikipedia feature scaling,” https://en.wikipedia.org/wiki/Feature_scaling, accessed: 2021-12-27.
- [349] M. L. Williams, P. Burnap, and L. Sloan, “Towards an ethical framework for publishing twitter data in social research: Taking into account users’ views, online context and algorithmic estimation,” *Sociology*, vol. 51, no. 6, pp. 1149–1168, 2017.
- [350] G. H. Winger, A. Oliver, J. Vičić, and A. Schaeffer, “Countering “fake news” through public education and advertisements: An experimental analysis,” *Political Research Quarterly*, vol. 77, no. 2, pp. 518–531, 2024.
- [351] A. K. Wood and A. M. Ravel, “Fool me once: Regulating fake news and other online advertising,” *S. Cal. L. Rev.*, vol. 91, p. 1223, 2017.
- [352] B. Wu, C. Xu, X. Dai, A. Wan, P. Zhang, Z. Yan, M. Tomizuka, J. Gonzalez, K. Keutzer, and P. Vajda, “Visual transformers: Token-based image representation and processing for computer vision,” *arXiv preprint arXiv:2006.03677*, 2020.

- [353] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, “Top 10 algorithms in data mining,” *Knowledge and information systems*, vol. 14, no. 1, pp. 1–37, 2008.
- [354] X, “Academic research access deprecated.” <https://twitter.com/TwitterDev/status/1641222788911624192>, accessed: 2023-03-30.
- [355] —, “Twitter,” <https://www.twitter.com/>, accessed: 2021-10-01.
- [356] —, “Twitter - country settings,” <https://help.twitter.com/en/managing-your-account/how-to-change-country-settings>, accessed: 2021-11-21.
- [357] —, “Twitter - rules and policies,” <https://help.twitter.com/en/rules-and-policies/notices-on-twitter>, accessed: 2021-11-19.
- [358] —, “Twitter - suspension rules,” https://blog.twitter.com/en_us/topics/company/2020/suspension, accessed: 2021-11-19.
- [359] —, “Twitter academic api,” <https://developer.twitter.com/en/products/twitter-api/academic-research>, accessed: 2021-11-05.
- [360] —, “Twitter badge,” <https://help.twitter.com/en/managing-your-account/about-twitter-verified-accounts>, accessed: 2021-10-16.
- [361] —, “Twitter tweet-lookup,” <https://github.com/twitterdev/Twitter-API-v2-sample-code/tree/main/Tweet-Lookup>, accessed: 2021-12-01.
- [362] —, “Twitter user-lookup,” <https://github.com/twitterdev/Twitter-API-v2-sample-code/blob/main/User-Lookup>, accessed: 2021-10-16.
- [363] —, “X username policy,” <https://help.twitter.com/en/managing-your-account/twitter-username-rules>, accessed: 2021-10-01.
- [364] —, “Twitter get-users,” <https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-users-search>, 2021, accessed: 2021-10-16.

- [365] ———, “X policies,” <https://help.twitter.com/en/rules-and-policies/twitter-impersonation-and-deceptive-identities-policy>, 2023, accessed: 2023-05-10.
- [366] E. Xing, S. Venkatraman, T. Le, and D. Lee, “Alison: Fast and effective stylometric authorship obfuscation,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 17, 2024, pp. 19 315–19 322.
- [367] B. Xu, R. Meng, Y. Chen, X. Xu, C. Dong, and H. Sun, “Latent semantic diffusion-based channel adaptive de-noising semcom for future 6g systems,” in *GLOBECOM 2023-2023 IEEE Global Communications Conference*. IEEE, 2023, pp. 1229–1234.
- [368] T. Xu, G. Goossen, H. K. Cevahir, S. Khodeir, Y. Jin, F. Li, S. Shan, S. Patel, D. Freeman, and P. Pearce, “Deep entity classification: Abusive account detection for online social networks,” in *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 4097–4114. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity21/presentation/xu-teng>
- [369] C. Yang, R. Harkreader, and G. Gu, “Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers,” in *Recent Advances in Intrusion Detection: 14th International Symposium, RAID 2011, Menlo Park, CA, USA, September 20-21, 2011. Proceedings 14*, vol. 8, 09 2011, pp. 318–337.
- [370] S. Yang, G. Huang, and B. Ofoghi, “Short text similarity measurement using context from bag of word pairs and word co-occurrence,” in *International Conference on Data Service*. Springer, 2019, pp. 221–231.
- [371] Y. Yang, B. Hui, H. Yuan, N. Gong, and Y. Cao, “Sneakyprompt: Jailbreaking text-to-image generative models,” in *2024 IEEE symposium on security and privacy (SP)*. IEEE, 2024, pp. 897–912.
- [372] M. Yousefi Kharaji, F. Salehi Rizi, and M. Khayyambashi, “A new approach for finding cloned profiles in online social networks,” *ACEEE International Journal on Network Security*, 06 2014.

- [373] J. Yuan, Y. Bian, X. Cai, J. Huang, Z. Ye, and K. Church, “Disfluencies and fine-tuning pre-trained language models for detection of alzheimer’s disease.” in *Inter-speech*, vol. 2020, 2020, pp. 2162–6.
- [374] K. Zarei, R. Farahbakhsh, and N. Crespi, “Deep dive on politician impersonating accounts in social media,” in *2019 IEEE Symposium on Computers and Communications (ISCC)*, 04 2019.
- [375] K. Zarei, R. Farahbakhsh, N. Crespi, and G. Tyson, “Impersonation on social media: A deep neural approach to identify ingenuine content,” in *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 10 2020.
- [376] S. Zhai, H. Chen, Y. Dong, J. Li, Q. Shen, Y. Gao, H. Su, and Y. Liu, “Membership inference on text-to-image diffusion models via conditional likelihood discrepancy,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 74 122–74 146, 2024.
- [377] C. M. Zhang and V. Paxson, “Detecting and analyzing automated activity on twitter,” in *PAM*, 2011.
- [378] C. Zhang, F. Pan, J. Kim, I. S. Kweon, and C. Mao, “Imagenet-d: Benchmarking neural network robustness on diffusion synthetic object,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 752–21 762.
- [379] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, “Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5907–5915.

- [380] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, “Pegasus: Pre-training with extracted gap-sentences for abstractive summarization,” in *International conference on machine learning*. PMLR, 2020, pp. 11 328–11 339.
- [381] N. Zhang, X. Mi, X. Feng, X. Wang, Y. Tian, and F. Qian, “Dangerous skills: Understanding and mitigating security risks of voice-controlled third-party functions on virtual personal assistant systems,” in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 1381–1396.
- [382] P. Zhang, G. Zeng, T. Wang, and W. Lu, “Tynllama: An open-source small language model,” 2024. [Online]. Available: <https://arxiv.org/abs/2401.02385>
- [383] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, “Vinvl: Revisiting visual representations in vision-language models,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 5579–5588.
- [384] R. Zhang, Z. Hu, H. Guo, and Y. Mao, “Syntax encoding with application in authorship attribution,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2742–2753.
- [385] S. Zhao, S. Grasmuck, and J. Martin, “Identity construction on facebook: Digital empowerment in anchored relationships,” *Computers in human behavior*, vol. 24, no. 5, pp. 1816–1836, 2008.
- [386] X. Zheng, Z. Zeng, Z. Chen, Y. Yu, and C. Rong, “Detecting spammers on social networks,” *Neurocomputing*, vol. 42, 02 2015.
- [387] A. Zhou, Y. Zhang, and M. Lu, “C-transformer model in chinese poetry authorship attribution,” *Int. J. Innov. Comput. Inf. Control*, vol. 18, pp. 901–916, 2022.
- [388] Y. Zhu, X. Liang, J. A. Batsis, and R. M. Roth, “Exploring deep transfer learning techniques for alzheimer’s dementia detection,” *Frontiers in computer science*, vol. 3, p. 624683, 2021.

- [389] —, “Domain-aware intermediate pretraining for dementia detection with limited data,” in *Interspeech*, vol. 2022. NIH Public Access, 2022, p. 2183.
- [390] F. Zollo, A. Bessi, M. Del Vicario, A. Scala, G. Caldarelli, L. Shekhtman, S. Havlin, and W. Quattrociocchi, “Debunking in a world of tribes,” *PLoS ONE*, vol. 12, 07 2017.