

Robert Cantaragiu

# **METAPRIV: ACTING IN FAVOR OF PRIVACY ON SOCIAL MEDIA PLATFORMS**

Master of Science Thesis  
Faculty of Information Technology and Communication Sciences  
Examiner: Assoc. Prof. Antonis Michalas  
January 2023

## ABSTRACT

Robert Cantaragiu: MetaPriv: Acting in Favor of Privacy on Social Media Platforms  
Master of Science Thesis  
Tampere University  
Masters Degree Programme in Information Security  
January 2023

---

Social networks such as Facebook<sup>1</sup> (FB) and Instagram are known for tracking user online behaviour for commercial gain. To this day, there is practically no other way of achieving privacy in said platforms other than renouncing their use. However, many users are reluctant in doing so because of convenience or social and professional reasons. In this work, we propose a means of balancing convenience and privacy on FB through obfuscation. We have created MetaPriv, a tool based on simulating user interaction with FB. MetaPriv allows users to add noise interactions to their account so as to lead FB's profiling algorithms astray, and make them draw inaccurate profiles in relation to their interests and habits. To prove our tool's effectiveness, we ran extensive experiments on a dummy account and three existing user accounts. Our results showed that, by using our tool, users can achieve a higher degree of privacy in just a couple of weeks. We believe that MetaPriv can be further developed to accommodate other social media platforms and help users regain their privacy, while maintaining a reasonable level of convenience. To support open science and reproducible research, our source code is publicly available online.

Keywords: Metaverse, Obfuscation, Online Profiling, Privacy, Social Networks, Recommendation Systems

Remark: A large part of this thesis has been accepted and published in a conference paper from SecureComm 2022, "MetaPriv: Acting in Favour of Privacy on Social Media Platforms" by R. Cantaragiu, A. Michalas, E. Frimpong, and A. Bakas.

The originality of this thesis has been checked using the Turnitin OriginalityCheck service.

---

<sup>1</sup>Since October 2021 is also known as META.

## **PREFACE**

Firstly, I would like to express my deepest gratitude towards Professor Antonis Michalas for guiding, supporting, and helping me throughout my thesis writing and degree studies. His course on Security Protocols was one of the best ones I took at the university and I managed to learn a lot of theoretical and practical skills from it. After the course, he offered me a position in his research group. There, he guided me through the process of writing academic papers and helped me get a deeper understanding of Information Security. His leadership and enthusiasm encouraged me to work harder and made me feel very grateful to work in the NISEC group.

I am also very thankful to Eugene Frimpong and Alexandros Bakas for helping and supporting me throughout my work in the research group. Their feedback and tips were very informative and made me a better researcher. They've helped me and contributed a lot in writing this thesis and other scientific papers that we've submitted together.

Finally, I would like to thank Billy Brumley, Juha Nurmi, and Marko Helenius for the wonderful and very interesting cybersecurity-related courses I took at the university. I enjoyed them a lot and I'm sure they will continue to inspire and bring many people into the field of cybersecurity.

Tampere, 1st January 2023

Robert Cantaragiu

## CONTENTS

1. Introduction . . . . .	1
2. Related work . . . . .	4
3. Preliminaries . . . . .	7
3.1 Facebook - the world's largest online social network . . . . .	7
3.2 Obfuscation . . . . .	8
3.3 Web automation . . . . .	8
4. System Model . . . . .	10
4.1 High-Level Overview . . . . .	10
4.2 Extending MetaPriv . . . . .	12
4.3 MetaPriv Denoiser . . . . .	13
5. Measuring User Privacy on Facebook . . . . .	15
6. Implementation and Results . . . . .	17
6.1 Dummy Account Results . . . . .	17
6.2 Privacy Results . . . . .	23
6.3 Real Account Results . . . . .	24
6.4 Limitations . . . . .	30
7. Use Case Protocol and Security Analysis . . . . .	32
7.1 Use Case Protocol . . . . .	32
7.2 Threat Model . . . . .	32
7.3 Security Analysis . . . . .	33
8. Conclusion and Societal Impact . . . . .	35
References . . . . .	36

## GLOSSARY OF ABBREVIATIONS

<i>c</i>	Ciphertext
<i>m</i>	Message
<i>t</i>	Timestamp
FB	Facebook
HTML	HyperText Markup Language
SN	Social Network
URL	Uniform Resource Locator
XSRF	Cross-site request forgery

## 1. INTRODUCTION

Internet privacy has become a major issue for many people in the current digital age [1, 2, 3, 4]. Concerns about online tracking and targeted advertising have only grown as a result of the popularity of social networks (SNs) and online tracking. In order to promote user engagement, these platforms make use of recommendation systems that display skewed information. They collect data about users' behavior, preferences, and interests and use this information to tailor their recommendations to each user's specific needs. For instance, when users share their opinions, beliefs, and preferences on social media platforms, e.g. by clicking 'like' on an article or by writing a controversial post, the recommendations they receive are aimed at reinforcing these beliefs. The goal of these systems is to provide users with information that most likely interests them and enables them to trace other users sharing the same values. However, this can create a cycle of subjectivity that can be difficult to break.

As users engage more and more with these platforms, they gradually become more entrenched in their existing beliefs, since the only information and news they receive affirms their already established opinions. This can lead to an echo chamber effect, where users are only exposed to information that affirms their existing beliefs. This can make it difficult for them to see other perspectives and engage in meaningful dialogue. It can also contribute to the spread of misinformation, as users are more likely to share and engage with information that is familiar to them, regardless of its accuracy. Moreover, social media platforms in collaboration with companies promoting their products manipulate user information for targeted advertising. Platforms use the data they collect to make accurate predictions about users' potential consumption needs, and then they provide this information to advertisers to enable them to tailor their ads to each user's specific interests and preferences. This can be problematic because it can lead to intrusive and manipulative advertising that targets users based on their personal information.

Targeted advertisements have several downsides, including being intrusive and annoying. Users are often bombarded with ads that are tailored to their interests and preferences, which can be frustrating and distracting. This can be especially problematic when ads are repetitive or seem to follow users across different websites and platforms. Targeted advertisements can also be manipulative and misleading, as advertisers can use data about users' online behavior to create ads that are designed to appeal to their emotions

and desires, rather than providing accurate and objective information about the products or services being advertised. Additionally, targeted advertisements can be discriminatory, as advertisers can use data about users' demographic characteristics to target ads to specific groups, which can reinforce stereotypes and perpetuate inequality. For example, if an advertiser only targets ads for high-paying jobs to men, this can lead to women being excluded from these opportunities.

*Balance between Privacy and Convenience on Social Networks:* Most users seem to be left with two options when it comes to social network privacy: (1) either regular use of the platform – hence no privacy or (2) complete abstinence from social networks – hence full privacy. However, the second option presents a number of problems:

- **Data removal:** The hassle of removing data about oneself from a platform, discourages users as it demands tedious action. Note that data removal does not refer to deleting the account alone, but to the deletion of all posts, pictures and logged data from the platform. Even in cases where all user data is deleted, SNs may still track users through partner companies on different websites (e.g. through FB Pixel [5])
- **Limited access to information:** Without access to social media, users may not be exposed to the same range of viewpoints, opinions, and news stories. This can limit their ability to stay informed about important issues and can make it difficult to engage in informed discussions and debates.
- **Social isolation:** Not using social media can lead to social isolation, as users may not have access to the same opportunities for connection and socialization. This can be particularly problematic for people who live in remote areas or who have limited opportunities to interact with others in person.
- **Missed opportunities:** Social media platforms can provide opportunities for networking, job hunting, and other forms of professional development. Without access to these platforms, users may miss out on important opportunities to advance their careers or expand their professional networks.
- **Difficulty staying in touch:** Without social media platforms, users may find it more difficult to maintain relationships with friends and family members that are geographically distant.
- **Reduced entertainment options:** There are a range of entertainment options, including games, videos, and other forms of content on these platforms. Without access to them, users may have fewer options for entertainment and may need to find other ways to occupy their time.

To this end, we believe that complete privacy is not achievable for most users. We do, however, think that one can strike a balance between privacy and convenience on said platforms and this has been a major motive behind our work. Our platform of choice for this work is FB – the world’s largest online SN. However, the idea presented below can be developed to accommodate privacy on other platforms.

Contributions: When it comes to privacy on social networks, we consider two forms of attacks:

1. External – privacy from other users i.e. how much information about ones profile is visible to other users. Fortunately, FB allows information hiding via private accounts and allowing users to change the visibility settings of their birthday, hometown, posts etc.
2. Internal – privacy from the social network itself. Every interaction (like, comment, post etc.) that one does on FB is monitored by the service to be later used in their profiling algorithms for targeted advertisement.

Our work focuses on proposing a solution for internal attacks and its main idea has been developed based on increasing concerns regarding the breach of user privacy in online SNs. More precisely, the main concern is that user choices are being covertly manipulated and controlled by SNs. With this in mind, we built *MetaPriv*, an automated tool that allows FB users to obfuscate their data and conceal their real interests and habits from FB. As a result, the core contribution of this paper is that it provides users with the necessary tools to protect their privacy when using SNs. It is worth mentioning that *MetaPriv* allows users to define the desired level of privacy (e.g. become almost 'invisible' online while still using SN platforms, reveal certain information about their digital and real lives etc.). By doing this, *MetaPriv* provides a novel and adaptive balance between privacy and functionality. This is a feature we believe will be used in several services in the near future.

## 2. RELATED WORK

User privacy protection on SNs has become a hot research topic in both academic and industrial communities due to the introduction of multiple governmental legislation aimed at protecting users from being exploited. To this end, a number of research works offer users a more private experience on Facebook and other platforms that are known to collect an abundance of user data.

FaceCloak [6] works by limiting Facebook's access to personal information. When a user creates a Facebook profile and fills in her personal information, including address, school, email, and so on, FaceCloak allows her to choose whether to display this information openly or to keep it private. If the user chooses to display the information openly, it is passed to Facebook's servers. If she chooses to keep it private, FaceCloak sends it to an encrypted storage [7] on a separate server. From there, it can be decrypted for and displayed only to friends who have authorization when they browse her Facebook page using the FaceCloak plug-in. Facebook never gains access to it. Other than that, FaceCloak generates fake information for Facebook's required profile fields, concealing from Facebook and from unauthorized viewers the fact that the real data is stored elsewhere. As FaceCloak passes the real data to the private server, it fabricates for Facebook a plausible fake person whose personal information such as gender, name and age, bears no relation to the real facts about the user. Under the cover of this plausible fake person, the user can forge real connections with her friends while presenting obfuscated data for others. The tool is implemented as a Firefox extension for FB. FaceCloak's user privacy attempt resembles our work. However, its main purpose is to hide specific data such as age, name, etc. and not user interests derived from interaction with the SN. Moreover, as of August 2011, the current version of the FaceCloak Firefox extension does not work with Facebook anymore due to changes made by FB [8].

The authors of Scramble [9] propose a system where users' data is scrambled before it is uploaded to the social network's servers. The system allows the social network to still perform essential functions, such as targeted advertising, while ensuring that users' data remains private. The scrambling technique is based on the mathematical concept of group theory. The authors use a type of group known as a "hidden subgroup" to encrypt social network data in a way that makes it unreadable to unauthorized third parties. The scrambling technique involves breaking up the data into small "chunks" and encrypting

each chunk separately. The encrypted chunks are then mixed together using a permutation function, making it difficult for anyone to determine the original order of the data. The resulting encrypted data is then uploaded to the social network's servers. To decrypt the data, authorized users use a key that enables them to reverse the permutation function and unscramble the data. The authors applied this technique to various types of social network data, such as friend lists and activity logs. In practice, the tool is an SN-independent Firefox extension allowing users to define access control lists (ACL) of authorised users for each piece of data, based on their preferences. In addition to that, it also allows users to encrypt their posted content in the SN, therefore guaranteeing confidentiality of user data against the SN. The tool allows users to hide information through cryptography, however this may require prior knowledge, which is usually counter intuitive for ordinary users. Also, it's implementation cannot be found anywhere and is likely outdated.

There are other privacy approaches that focus on different platforms: Google, Youtube, Amazon etc. While they do not necessarily provide solutions for achieving privacy on Facebook, their approaches inspired our work. For example, *TrackThis* [10] is an experimental project by Mozilla that provides a privacy-focused browsing experience by generating a large number of fake web browsing activities to confuse online trackers. It generates fake web browsing activities, such as searches, visits to websites, and clicks on links, to make it difficult for trackers to differentiate between your actual activity and the fake activity. *TrackThis* is not a browser extension or add-on, but rather a web-based tool that allows one to select a profile, such as "hypebeast," "filmmaker," or "vegetarian," and then generates a series of fake web browsing activities based on that profile. The tool is intended to demonstrate the impact of online tracking and the benefits of using privacy-enhancing tools. Similarly, the authors of [11] and [12] show a way to attack personalization algorithms by polluting a users browser history with noise through Cross-Site Request Forgery (CSRF), clickjacking and cross-site scripting (XSS). In [13], the authors present an attack for draining ad budgets. By repeatedly pulling ads using crafted browsing profiles, they managed to reduce the chance of showing their ads to real visitors and trash the ad budget. While having similar approaches to ours, these tools provide limited privacy in the long run as they have to be relaunched after a period of time.

In [14], the authors test protesting against data labouring [15]: they utilize user interactions with different services as input for training user profiling algorithms. They simulate data strikes against recommendation systems under various conditions. To evaluate the effectiveness of these data strikes, the authors conducted a series of experiments using Amazon Mechanical Turk. Participants were asked to participate in a data strike against a hypothetical social networking site called "Friendbook." The participants were given different levels of information about the data strike and were asked to indicate whether they would participate. The results of the experiments showed that participants were more

likely to participate in a data strike when they were given more information about the data practices of the company. However, the authors also found that the effectiveness of a data strike depended on the size of the user base and the level of dependence on the service. Their results imply that data strikes can put a certain pressure on technology companies and that users have more control over their relationship with said companies. Our work can also be viewed as a protest against the data labouring of users on an SN: if enough users had access to noise attributes, the recommendation systems of FB would most likely be disrupted even for new users not using our tool.

Howe and Nissenbaum proposed a tool called AdNauseam [16] – a free browser extension designed to obfuscate browsing data and protect user-tracking by advertising networks. The tool blocks and "clicks" on online ads in a way that makes it difficult for advertisers to track users or collect personal information. The extension also uses a unique approach to "protest," in that it clicks on every ad it encounters, thereby generating false click-through data that can be used to undermine the effectiveness of targeted advertising. The authors of HARPO [17] propose a novel approach to subverting online behavioral advertising (OBA) using machine learning. They argue that OBA poses a threat to users' privacy and can be used to manipulate user behavior. They propose a system called HARPO that is designed to subvert OBA by generating "fake" user profiles that mimic real users but contain false information. The system uses a machine learning algorithm to analyze the advertisers' tracking data and generate profiles that are difficult for the advertisers to distinguish from real users. The authors also describe how HARPO can be used to disrupt targeted advertising by generating large numbers of fake clicks on ads. HARPO is also able to achieve better stealthiness to adversarial detection as compared to AdNauseam. Our tool is designed and based on similar obfuscation ideas, however we focus on a specific SN platform and not only on advertisements.

MetaPriv is designed to accommodate users who want to obtain a higher level of privacy when using FB. It achieves this by trying to confuse the Social Networks profiling algorithms, since these rely on the user interactions with the platform. The idea is very similar to what AdNauseam and HARPO does, however these tool focuses specifically on advertisements that are shown on multiple websites. It does not take into account user interactions with FB such as liking, commenting or reacting to a FB post.

## 3. PRELIMINARIES

### 3.1 Facebook - the world's largest online social network

The largest online social network to date is Facebook having an average of 1.88 billion daily active users and 2.85 billion monthly active users as of March 2021 [18]. Facebook's business strategy relies on serving its users advertisements from third parties that wish to further promote their products and services. These companies, in turn, pay Facebook to have their products promoted to the people who are most likely to buy them. This motivates the social network to keep their users engaged while learning as much information as possible about them. This allows Facebook to measure more accurately the likelihood of a person clicking on an ad.

Facebook is used for sharing interests, opinions, watching videos, following news and so on. Other than that, it is part of a larger company that also owns Instagram, WhatsApp and has numerous contracts with third-party websites and apps which allows it to collect information about a persons browsing habits. With all this data, the company can make an accurate profile of an individual, and by recognising and analysing patterns with similar individuals, it can predict a users preferences, habits and interests.

A large amount of data collection however, can always lead to privacy concerns and violations. Some examples specific to Facebook include:

- Cambridge Analytica [19] – in 2014 the personal data of up to 87 million FB users was harvested without their consent, by exploiting their friendship connection to the users who sold their data via the FB app.
- Onavo [20] – in February 2018, it was reported that FB had begun to include advertising for the Onavo Protect app within the FB app for iOS users in the United States. This led to denouncements of the app by media outlets, who classified Onavo as spyware because it is used by FB to monetize usage habits within a privacy-focused environment, and because the app listing did not contain a prominent disclosure of FB's ownership.
- On June 7, 2018, FB announced that a bug had resulted in about 14 million FB users having their default sharing setting for all new posts set to "public". [21]

To this end, it is easy to see how some users have grown to mistrust FB and seek to get

more privacy while using the service.

### 3.2 Obfuscation

Obfuscation refers to the act of intentionally making something difficult to understand or obscure, often by deliberately making it unclear, confusing, or complex. Generally, it has multiple definitions in the context of Information Technology and Digital Privacy:

- In software engineering, the term obfuscation is used to refer to making the source code of a program as hard as possible to recover to a human readable state.
- In network and anonymity technologies, it refers to methods to obscure data from inspection by network protection systems. For example, there is `obfs4proxy` [22] – a tool that attempts to circumvent censorship by transforming the Tor traffic between the client and the bridge. Its purpose is to keep a third party from telling what protocol is in use based on message contents.
- In the context of privacy against data collectors such as a social network, obfuscation involves hiding legitimate data in an abundance of similar data that serves as noise to confuse a curious data analyst [23].

For this work, the last definition is of particular interest to us, as it aligns with the goal of confusing social networks. With this in mind, we propose a way for users to preserve their privacy on social networks using obfuscation. The method is based on the idea that since these services collect data about a user, why not give them also an abundance of random data (noise) to collect from the user's account.

### 3.3 Web automation

Web automation refers to the process of automating tasks on the internet or web using software tools or scripts. It involves the use of software programs, known as web automation tools, to perform tasks that would otherwise require manual input or intervention. These tools can be used to automate tasks such as data extraction, form filling, web scraping, website testing, and repetitive tasks that would typically be performed by a human. The automation process involves creating scripts or programs that interact with web browsers and web applications to perform these tasks.

Web automation can improve productivity and efficiency by reducing the time and effort required to complete routine tasks. It is commonly used in businesses and organizations to automate tasks such as data entry, customer support, and inventory management, among others. Some examples of web automation frameworks include: Selenium, Gauge, Serenity etc.

For this work, we take advantage of the Selenium framework. Selenium is widely used in

software testing for automation of functional and regression testing of web applications. It supports different types of web elements and provides various methods for interacting with them, like clicking buttons, filling in forms, selecting checkboxes, and many more. It also provides advanced features like handling pop-ups, windows, alerts, and frames, among others. MetaPriv uses it to automate user interaction with Facebook through a webdriver instance of Firefox. A webdriver is a remote control interface that allows control of user agents. It provides a way for out-of-process programs, such as python, to remotely instruct the behavior of web browsers [24].

## 4. SYSTEM MODEL

We now proceed with introducing the system model we consider by describing the main entities participating in the design of MetaPriv, as well as their capacities.

**Social Network (SN):** Defined as a graph  $\mathcal{G} = (\mathcal{U}, \mathcal{R})$  where the vertices are comprised of users from a set  $\mathcal{U}$ , with the edges being the relationship between said users, described by the set  $\mathcal{R} \subseteq \{\{u, v\} \mid u, v \in \mathcal{U} \text{ and } u \neq v\}$ .

**Users:** Let  $\mathcal{U} := \{u_1, \dots, u_n\}$  be the set of all users registered in an online SN such as FB. Each user has a unique identifier  $i \in [1, n]$ . In addition to that, each user is associated with a number of attributes. The set of all attributes associated with a user  $u_i$  is denoted as  $\mathcal{A}_i \subseteq \mathcal{A}$ .

**Attributes:** The set of all available attributes in an SN is denoted by  $\mathcal{A} := \{a_1, \dots, a_m\}$  and is called the attribute space. An attribute is a specific trait that a user  $u_i$  possesses, e.g. “ $u_i$  likes cats”.

**BOT:** An entity that adds noise to a user profile ( $u_i$ ). It works by mimicking the user’s interaction with the SN and generates noise attributes on their behalf.

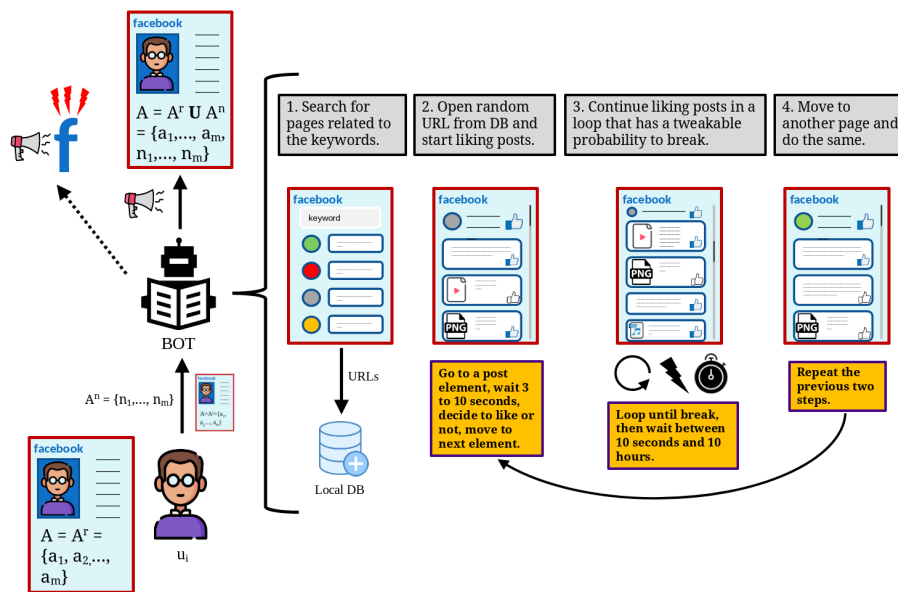
**User Real and Noise Attributes:** Assume a user  $u_i$  with a list of attributes  $\mathcal{A}_i$ . Elements of  $\mathcal{A}_i$  may have been generated legitimately (i.e. through the user’s real activity) or by the BOT. The set of all attributes generated by the user’s legitimate activity is denoted as  $\mathcal{A}_i^r \subseteq \mathcal{A}_i$  while the set of all attributes associated with  $u_i$  but generated by the BOT is denoted by  $\mathcal{A}_i^n \subseteq \mathcal{A}_i$ .

### 4.1 High-Level Overview

The core idea behind MetaPriv is to fuddle FB’s opinion about a user  $u_i$  by obfuscating  $u_i$ ’s real attributes  $\mathcal{A}_i^r$  with the help of noise attributes  $\mathcal{A}_i^n$ . To that end, we use the BOT and have it interact with the SN on behalf of  $u_i$ . Ideally, to achieve privacy, the amount of traffic generated by the BOT should be the same or more than the traffic generated by  $u_i$ .

When user  $u_i$  creates an account on FB, they have no attributes (i.e. the set  $\mathcal{A}_i$  is empty). Following registration,  $u_i$  begins generating activity (e.g. adding friends, liking pages and

posts). By collecting and analyzing user activities, FB creates a list of attributes that represents each user's perceived interests (e.g.,  $a_1$  – “ $u_i$  likes cooking”). For the purposes of this work, we consider these attributes as real and are added to the set  $\mathcal{A}_i^r$  – a subset of  $\mathcal{A}_i$ , i.e.  $\mathcal{A}_i^r \subseteq \mathcal{A}_i$ . The set  $\mathcal{A}_i$  is then used by FB to decide which posts and advertisements are presented in the respective  $u_i$  feed. In this scenario, all  $u_i$ 's interests are known to the SN, which can make accurate predictions about their preferences and therefore populate their account with accurate personalized content. In this work, we are examining ways of protecting user privacy from a potentially malicious or at least curious SN. To achieve this, we have created MetaPriv. With our tool, users can confuse an SN about their real interests. MetaPriv revolves around a simple idea: Since the SN personalizes users by analyzing their activities on the platform, our tool generates noise traffic on behalf of a user. This will result in adding attributes to the set  $\mathcal{A}_i^n$  containing the noise attributes described earlier. With this in mind, we built a BOT as part of the core of MetaPriv whose functionality is described below. At this point, it is worth noting that the interactions generated by MetaPriv consists of primarily liking posts and pages.



**Figure 4.1.** High-level overview of the BOT's functionality.

1. As a primary requirement, the BOT needs access to  $u_i$ 's account. This can be done in one of two ways: Either with  $u_i$  providing their credentials or through their browser profile folder i.e. the hidden folder in an operating system's user folder, where all web browser cookies, etc. are stored.
2. Once the BOT has gained access to the user account, it requires a set of keywords generated by a different part of MetaPriv, which would serve as noise attributes. The keyword generator, however, requires a seed keyword that the user must input at least once.
3. The user then inputs their desired level of privacy. This privacy level simply refers

to the level of convenience and benefits that a user is willing to accommodate to better protect their privacy. In practice, it represents the amount of noise that is persistently added to an account.

4. Finally, the BOT repetitively executes a series of steps represented in Figure 4.1.

## 4.2 Extending MetaPriv

After extensive experiments, we observed limited success with the initial version of MetaPriv, which we attributed to its limited interactions (i.e., simply liking posts and pages). These results are discussed in chapter 6. As such, it became necessary to add extra features to MetaPriv. To limit the amount of noise generated, before the BOT switches to another page, it waits for a random amount of time. In the basic implementation, MetaPriv did not run any tasks during this wait period. However, in this extended version, MetaPriv watches keyword related videos and clicks Facebook ads displayed in user's main feed instead of simply waiting. Our observations showed that video watching did not seem to raise any suspicions from FB, i.e. the browser session did not get logged out or blocked, hence the BOT clicks on every ad from the first 100 posts in the main feed, searches the keyword in FB's video page and watches all the videos returned. This, we believe, helps to further reinforce the noise and give it more variety. Figure 4.2 provides an overview of the extended functionalities of MetaPriv.

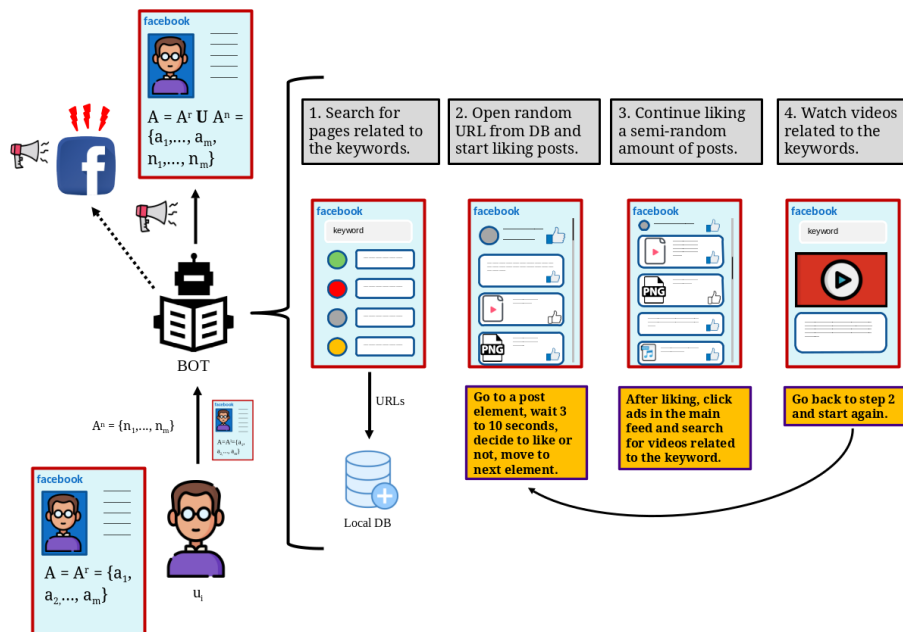
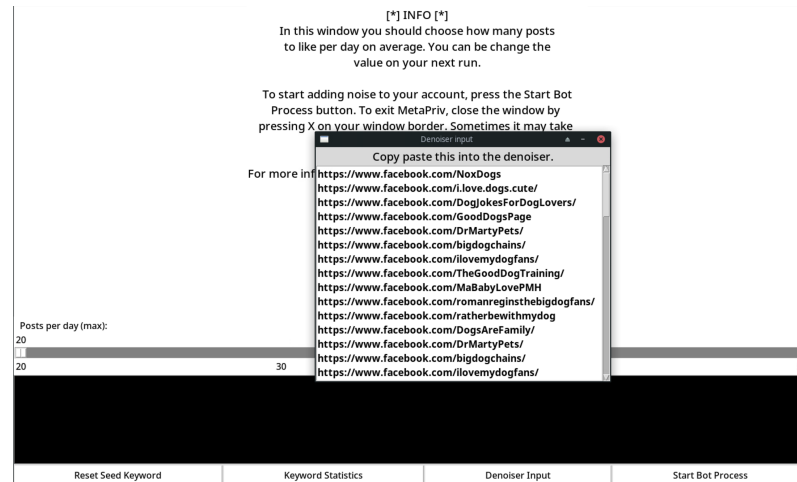


Figure 4.2. Extended BOT Functionality.

### 4.3 MetaPriv Denoiser

Although MetaPriv achieved its intended purpose of confusing the SN with regards to the user's interests, it soon became obvious that users may find their FB feeds to have become unappealing, as posts appearing on them are of no interest. Additionally, using MetaPriv seems to attract a high amount of noise feed, based on the feedback we've received about the tool. This may cause user hesitation regarding MetaPriv.

To solve this issue, we introduced a simple noise filtering tool implemented as a browser extension<sup>1</sup>. As previously described in section 4.1, MetaPriv saves the URLs of the FB pages used for generating noise. These URLs can be found in a local database that only the user can access (Figure 4.3). Additionally, this database contains all the keywords used by MetaPriv. With this in mind, we can use these two datasets to decide which posts to filter from FB feeds.



**Figure 4.3. Denoiser Input**

Generally, FB shows two types of posts in a user's feeds: (1) posts from pages that the user already liked and (2) recommended/suggested posts from other pages. The first one is very straightforward to filter – since we know which pages are noise related from the database, we just remove the posts that come from these pages in the FB feeds. For the second type, we need to filter based on the information describing the FB post. This information consists of text in the description of the post and the page name. This is the only data that can be extracted from the post element in the HTML code of the page and it can easily be found in the inner HTML of the element. Using this text, we can check if a noise keyword is present in it and filter the post.

Filtering based on words, however, has limitations. First, it can lead to false positives where a post gets filtered even if unrelated to the noise or if related to both the noise and real interests. For example, if a user likes Apple, hates Samsung and has Samsung as

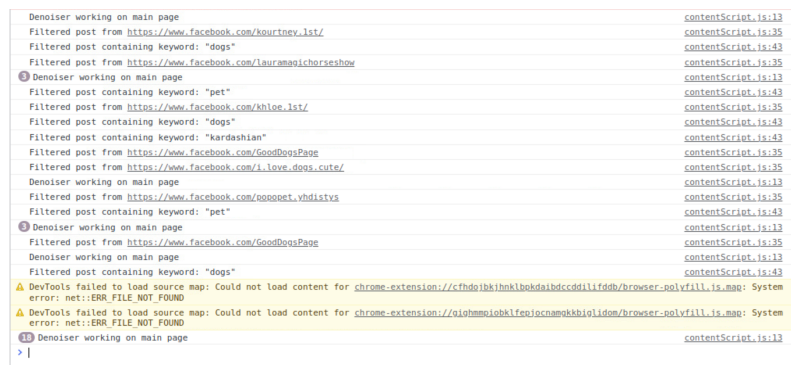
<sup>1</sup>[https://github.com/ctrgrb/MetaPriv\\_denoiser](https://github.com/ctrgrb/MetaPriv_denoiser)

a noise keyword, the tool can filter posts like "Top 10 best phones of 2022", which could include both keywords. False negative are another limitation where the tool cannot filter posts clearly related to the noise, which, however, do not contain that specific keyword. For example, if the user's noise keyword is "football", it may not be able to filter a video post from the page "FC Barcelona", which, despite its description, clearly depicts a football game. This issue can potentially be resolved by filtering related keywords. However, more keywords would also generate more false positives.



**Figure 4.4.** Denoiser Web Extension

Finally, it is also worth mentioning that users can simply opt out of filtering based on words, as the tool can work perfectly well by filtering page URLs. Figure 4.5 shows the output when the denoiser is run on the home page of one of the user accounts we used for experiments. The output shows the denoiser filtering posts related to the keywords submitted to the tool (Figure 4.4).



**Figure 4.5.** Denoiser Output

## 5. MEASURING USER PRIVACY ON FACEBOOK

Previous works focus on measuring privacy according to the visibility and sensitivity of user attributes [25, 26, 27, 28]. This approach, however, is inapplicable, as the aim is to confuse the data collector, thus leading to inaccurate user profile predictions. Visibility of a user’s attributes would always be maximum, since the SN stores all user interactions with it. Additionally, in this work the concept of sensitivity cannot apply, since all user attributes are known to the SN (i.e. can be considered public). With this in mind, we propose a new definition for privacy on an SN based on a user’s *real* and *noisy* interactions with the SN. Real interactions are daily, *legitimate* user interactions with the SN. Noisy interactions are BOT-produced and mainly generate fake activity on a user’s profile.

Our first approach on quantifying privacy was characterized by rather elementary and naive thinking: Initially, we defined the notion of *Theoretical Privacy*. The intuition behind Theoretical Privacy was that a user’s level of privacy is proportional to the number of noise in their profile. However, the results of our first experiments did not support this. Apparently, the time that a user likes a post, a page, etc. seems to be significant for FB’s personalization algorithms. More precisely, it seems that FB weighs a user’s recent rather than older content. In view of the above, we refined our idea on quantifying privacy and defined *Effective Privacy* – an alternative that better fits FB’s models.

**Definition 1** (Theoretical Privacy). *Theoretical privacy is measured by taking into account the amount of posts liked by a user  $u_i$  and the BOT. User  $u_i$ ’s theoretical privacy with  $j + k$  attributes is defined as:*

$$P_i^{th} = \frac{\sum_{j \in \mathcal{A}_i^r} RA_j^{th} - \sum_{k \in \mathcal{A}_i^n} NA_k^{th}}{T}, \quad (5.1)$$

where  $RA_{th}$  is the number of specific attribute-related posts liked by  $u_i$ ,  $NA_{th}$  is the number of specific attribute-related posts liked by the BOT and  $T$  is the total number of posts liked by  $u_i$ ’s account.

**Definition 2** (Effective Privacy). *For this definition we consider the effective strength of user real and noise attributes. The strength of a user’s real attribute is proportional to:*

- the number of posts in the main feed from liked pages linked to an attribute. Vari-

able:  $r_p$

- the number of recommended, suggested and sponsored posts in the main feed from pages linked to an attribute, but not liked by the user or the BOT. Variable:  $r_{sp}$
- the number of video posts from the main video feed (<https://www.facebook.com/watch>) linked to an attribute. Variable:  $r_{vp}$
- the number of video posts from the latest video feed (<https://www.facebook.com/watch/latest>) linked to an attribute. Variable:  $r_{lvp}$

The effective strength of a real attribute is defined as:

$$RA_{eff} = \frac{1}{n} \left( a \frac{r_p}{t_p} + b \frac{r_{sp}}{t_{sp}} + c \frac{r_{vp}}{t_{vp}} + d \frac{r_{lvp}}{t_{lvp}} \right), \quad (5.2)$$

where  $a, b, c, d \in \{0, 1\}$ ,  $n = a + b + c + d$ ,  $t_p$  is the total number of posts shown in the main feed,  $t_{sp}$  is the total number of suggested posts shown in the main feed,  $t_{vp}$  is the total number of video posts related to  $u_i$ 's attributes from the main video feed and  $t_{lvp}$  is the total number of video posts from the latest video feed. Each of the variables  $a, b, c, d$  is given the value 0, when their respective fraction is 0. Otherwise they are given the value 1. This is done so that, if one effective strength variable has a value of 0 (i.e. no posts), then it will not be taken into account for the final effective privacy value.

A similar definition stands for the effective strength of noise attributes  $NA_{eff}$ . variables  $r_p, r_{sp}, r_{vp}$  and  $r_{lvp}$  are replaced with corresponding noise attributes i.e.  $n_p, n_{sp}, n_{vp}$  and  $n_{lvp}$ . The strength of a noise attribute is defined as:

$$NA_{eff} = \frac{1}{n} \left( a \frac{n_p}{t_p} + b \frac{n_{sp}}{t_{sp}} + c \frac{n_{vp}}{t_{vp}} + d \frac{n_{lvp}}{t_{lvp}} \right) \quad (5.3)$$

Finally, for a user  $u_i$  with  $j + k$  attributes, we combine the two variables and reach the effective privacy:

$$P_i^{eff} = \sum_{j \in \mathcal{A}_i^r} RA_j^{eff} - \sum_{k \in \mathcal{A}_i^n} NA_k^{eff} \quad (5.4)$$

In both cases, the resulting value will be  $P \in [-1, 1]$ . The closer it is to 0, the more indistinguishable will the noise attributes be from real attributes. Therefore, the account of an arbitrary user  $u_i$  is private iff  $P \approx 0$  or  $P \leq 0$ .

## 6. IMPLEMENTATION AND RESULTS

To demonstrate MetaPriv's functionality and practicality, we evaluated both the basic and the extended versions. For the basic version, we created a dummy FB account and ran a 10-week experiment to build the account's real and noise attributes. While for the extended version, we tested MetaPriv on two real FB accounts that have existed and are active for over a decade. To evaluate the dummy account, we used MetaPriv to simulate both user and BOT interactions<sup>1</sup> with FB. Our test program was implemented using Python 3.10 and Selenium WebDriver – a framework for testing web applications that allowed us to simulate an automated user interaction with FB.

***Open Science and Reproducible Research:*** Our source code<sup>2</sup> has been made publicly available online to support open science and reproducible research. Interested reviewers can also download our application for possible testing or a simple overview of the generated research artifact. The tool is designed as a python program that can work on Linux devices and requires a number of libraries to function. On first launch, it opens up a GUI that asks the user to input the level of his desired privacy, a keyword, her FB credentials and a password used to encrypt her data in the local database. Then, it open a headless web-browser instance and starts generating noise to the users account. Snapshots and logs are shown to the user in the same GUI so that she may observe the BOT's functionality. On the next launches, the credential and keyword inputs are not necessary since the tool will make a new browser profile folder which it will use for gaining access to the account. Also, new keywords are generated by the tool for a more automatized experience. Finally it is worth noting that the tool can be used by running it continually on a home server or by running it on a portable device such as a laptop. In case the device will drop its network connection or suspend, the tool will automatically pause and resume after the connection is restored or the device is awoken.

### 6.1 Dummy Account Results

For the dummy FB account, we created a 22-year-old female user from Ireland (the account and all interactions were made through an Azure server with an Irish IP address).

---

<sup>1</sup>We make a clear distinction between MetaPriv and the BOT. BOT interactions will be used to refer to the noise traffic generated by MetaPriv.

<sup>2</sup><https://github.com/ctrgrb/MetaPriv>

At the end of each week, we ran an extensive analysis of FB's main, video and latest video feed by opening the respective URLs, going through a certain amount of posts in them and saving the information about said posts in an SQL database.

**Weeks 1 & 2:** The first two weeks primarily consisted of building the user profile with a single attribute. To be more specific, we used the attribute "cat", so FB would associate our user with cats. We then provided the keyword "cat pictures" as input to MetaPriv. The program liked 1,056 posts from 51 keyword-related pages over these two weeks. This keyword served as the user's *real attribute*. After one week, 'Recommended' posts appeared in the main feed. Out of 264 posts, 32 were recommended and 11 seemed relevant to the user's profile:

1 post related to demographics -a house in Dublin; 1 post about cats from a page about cats; 2 posts about tigers (both from FB group: WildCat Ridge Sanctuary); 1 post about demographics and cats (page name: North Dublin Cat Rescue Ireland); 1 post about ostriches, 1 about bulls, 2 about dogs, 1 about rare animals (related to animals); 1 post about "Dads Acting Like Their Teenage Daughters" (possibly gender-related).

Other recommended posts were unrelated to "cats" and had a dozen million views (we assume these were most likely trending posts). Almost all the recommended posts were videos<sup>3</sup>. After these two weeks, we analyzed 449 posts from the main feed and got 13 recommended posts along with 23 "join group" recommendations from cat-related FB groups. 8 of the recommended posts were linked to the user's profile:

1 post related to demographics: Football game GERMANY vs IRELAND (2002); 1 post about cats from FB group: CAT LOVERS PHILIPPINES; 4 posts about animals from a group about animal comics; 1 post about cats from the 'Daily Mail' page; 1 post from a group about Dinosaurs. The name of the person posting was: Margaret Happycat.

This time, most recommendations appeared from groups, though the user was not a member of any.

**Week 3:** For the third week, we added a second keyword as a noise attribute to the profile. At this point, the noise was manifested through liking a noise-related page and its posts at every 10th page switch. In essence, 10% of the interactions with FB were now related to a single noise attribute. This 10% represented 72 out of 554 posts liked in week 3 from 5 pages linked to the chosen noise keyword "guns"<sup>4</sup>. We observed that there were no recommended posts after this period. An analysis of 547 posts from the main feed showed that 19 were linked to the noise attribute. The latest video feed contained only 21

<sup>3</sup>This could be because users show a higher rate of engagement to online videos compared to text (e.g. articles, blog posts, etc.)

<sup>4</sup>It is worth noting that the percentage value is an approximation since MetaPriv is designed with randomness in mind to avoid patterns in its behaviour.

videos from liked pages related to the real attribute (i.e. cats). In the main video feed, we analyzed 184 video posts. 70 of them included words such as: ['cat', 'Cat', 'kitten', 'Kitten'] in their description or page URL and were, thus, related to the real attribute, while nothing was related to the noise attribute.

**Week 4:** For this period, we increased the noise amount from 10% to 20%. Out of 530 liked posts, 112 came from 8 pages related to the noise attribute. In the main feed, out of 337 posts, 38 were from pages related to the noise attribute. FB stopped showing recommended posts at this point, however, 'Suggested for you' posts began to show. Out of the 337 posts, 8 were labeled as 'Suggested' out of which 1 was related to animals, 3 specifically to cats and the remaining were possibly gender-related. This time too, the latest video feed showed only cat-related videos and in the main video feed, out of 152 videos, 35 included the words: ['cat', 'Cat', 'kitten', 'Kitten'] in the description or page URL, while no videos were related to guns.

**Week 5:** We decided to add another noise attribute, thus dividing FB interaction as follows: 70% cats, 20% guns and 10% cooking. From a total of 485 liked posts, 130 were related to the keyword "guns" and 36 to "cooking recipes". This time, out of 673 posts in the main feed, 67 were related to guns and 147 to cooking. Our theory for increased cooking content is that a cat lover is more likely to also like cooking rather than guns<sup>5</sup>. This time, out of 16 suggested posts, 14 were cats. In the latest video feed, out of 51 videos, 21 were cats, 1 guns and 26 cooking. Finally, in the main video feed, out of 136 posts, 27 were cats, 3 guns and 7 cooking.

**Week 6:** We increased the amount of noise for the cooking attribute to 20% and the gun attribute to 30%, thus dividing FB interaction as follows: 50% cats, 30% guns and 20% cooking. From a total of 647 liked posts, 213 were guns and 125 cooking. In the main feed, out of 405 posts, 35 were guns and 66 cooking. There were also 7 suggested posts, out of which 4 were cooking and 2 cats. In the latest video feed, out of 65 posts, 12 were cats, 2 guns and 51 cooking. Finally, in the main video feed's 103 posts, 27 were cats and 15 cooking.

**Week 7:** We added another noise attribute that would be stronger than others. Hence, FB interaction became: 23% cats, 23% guns, 23% cooking and 30% chess. From a total of 365 liked posts, 90 were cats, 89 guns, 76 cooking and 110 chess. The main feed's 286 posts were divided as follows: 45 guns, 72 cooking and 2 chess. From 14 suggested posts, 10 were cooking and 1 chess. In the latest video feed, out of 162 posts, 18 were cats, 35 guns, 83 cooking and 22 chess. The 137 posts in the video feed were divided as follows: 25 cats, 1 guns, 9 cooking and 1 chess.

**Week 8:** The aim was to examine results, when new attributes were added without re-

---

<sup>5</sup>This might also be related to the fact that Ireland has one of Europe's least permissive firearm legislation – hence gun-related content is heavily regulated.

inforcing old ones. For the first half of the week FB interaction was 100% fishing-related and the second half 20% fishing and 80% bodybuilding.

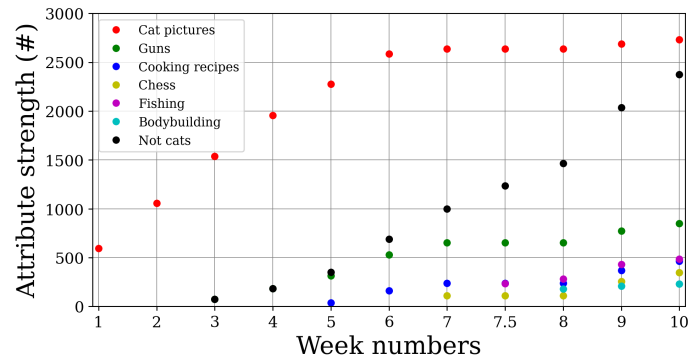
- First half: Liked 235 posts about fishing. In the main feed, out of 402 posts, 207 were cats, 45 guns, 115 cooking, 4 chess and 15 fishing. Out of 7 suggested posts, 4 had to do with fishing and the others were unrelated to the user's attributes. In the latest video feed, from 190 videos, 14 were cats, 48 guns, 72 cooking, 39 chess and 18 fishing. In the main video feed, out of 148 videos, 12 were cats, 2 guns, 10 cooking, 3 chess and 1 fishing.
- Second half: Liked 48 fishing posts and 181 bodybuilding posts. In the main feed, out of 423 posts, 229 were cats, 33 guns, 127 cooking, 22 fishing and 7 bodybuilding. Out of 2 suggested posts, 1 was bodybuilding and the other unrelated. In the latest video feed, out of 156 videos, 16 were cats, 9 guns, 30 cooking, 34 fishing and 72 bodybuilding. In the main video feed, out of 128 videos, 1 was cats, 2 guns, 20 cooking, 1 chess and 1 fishing.

**Week 9:** We ran MetaPriv with 10% cat-related traffic and the remaining with the following noise attribute layout: 20% guns, 20% cooking, 20% chess, 20% fishing, 10% bodybuilding. From 626 liked posts, 51 were about cats, 122 guns, 130 cooking, 144 chess, 149 fishing and 29 bodybuilding. In the main feed, out of 460 posts, 199 were about cats, 51 guns, 145 cooking, 19 chess, 25 fishing and 7 bodybuilding. This time there were no suggested posts. In the latest video feed, from 154 videos, 18 had to do with cats, 14 guns, 77 cooking, 35 chess and 18 fishing. In the main video feed, from 137 videos, 25 were about cats, 1 guns, 9 cooking and 1 chess.

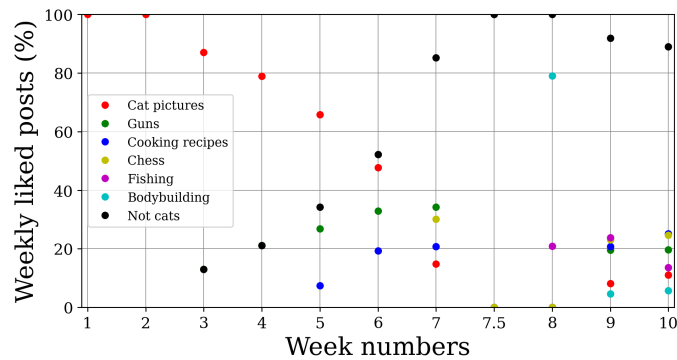
**Week 10:** In the last week we ran MetaPriv with the same parameters as in week 9: 10% cats, 20% guns, 20% cooking, 20% chess, 20% fishing and 10% bodybuilding. From 381 liked posts, 42 were cats, 75 guns, 96 cooking, 94 chess, 52 fishing and 22 bodybuilding. In the main feed, out of 442 posts, 160 were cats, 71 guns, 139 cooking, 30 chess, 32 fishing and 4 bodybuilding. Again, there were no suggested posts. In the latest video feed, from 133 videos, 10 were cats, 15 guns, 75 cooking, 22 chess and 12 fishing. Finally, in the main video feed, from 124 videos, 6 were cooking, 1 chess and 2 bodybuilding.

The total amount of posts liked on a weekly basis for each attribute (attribute strength), is shown in 6.1a. The week number is noted on the horizontal axis and the attribute strength (total amount of posts liked) on the vertical axis. As the figure indicates, even on week 10, the 'cat' attribute strength outweighs all others combined, since the attribute remained reinforced even when said reinforcement decreased over time. 6.1b represents the ratio of posts. Here, the ratio is calculated using the posts liked on a specific week, omitting those of previous weeks. This time, the attribute strength on the vertical axis stands for the percentage of liked posts for each attribute.

Next, we present the results of each variable for the effective attribute strength. The



(a) Weekly progression of theoretical attribute strength



(b) Ratio of weekly liked posts

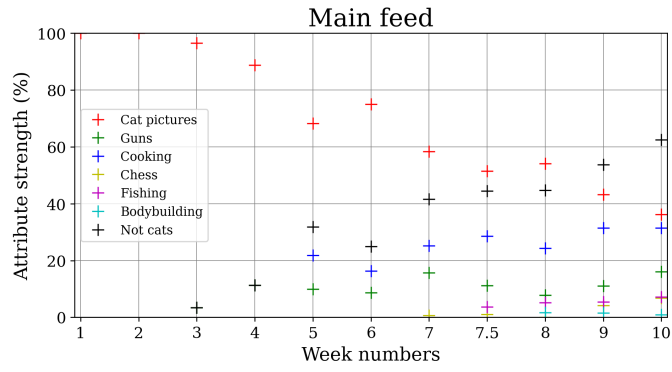
**Figure 6.1.** The total amount of posts liked and the ratio of posts liked per week.

main feed, recommended posts, latest video feed and main video feed are represented in Figure 6.2.

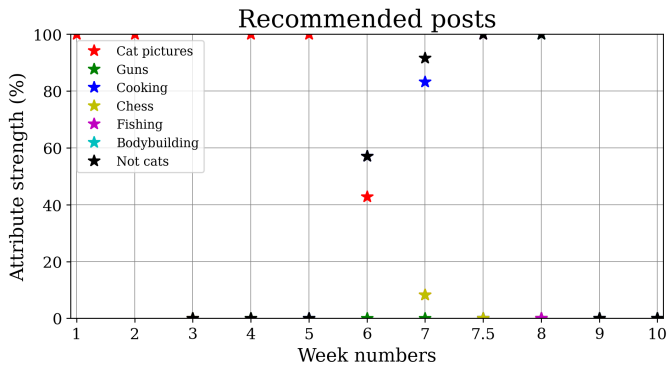
We can, now, compare results between Figure 6.2 and 6.1b: on weeks 5 to 8, noise-effective attribute strength variables approached real variables. 6.1b shows that around week 6, there are more noise-related likes than real likes. Consequently, FB's recommendations show more noise-related content as we can see from Figure 6.2. In the first 4 weeks, 6.2c and 6.2d show no relation to noise attributes. We thus conclude that 20% noise is not enough to change said variables. Also, 6.2b shows that in a few weeks' time, there were no recommended/suggested posts in the main feed (weeks 3, 9 and 10).

To avoid confusion in Figure 6.2, we must clarify that in the main video feed 6.2d and in the recommended, suggested and sponsored posts 6.2b, the FB content is derived from pages not liked by the user. The content is both user attribute-related and unrelated. It is assumed that the unrelated content is presented by FB because of other features in their recommendation systems e.g. users who liked X also liked Y. Their recommendation algorithms are not open source, hence their mode of operation is concealed. Due to this, our results are based on content exclusively related to user attributes.

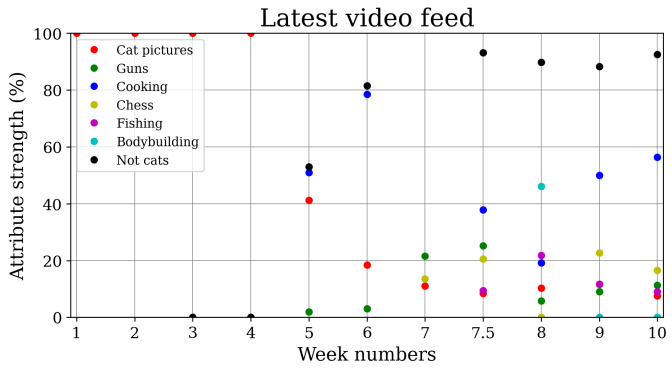
Following, we present our privacy results per week and discuss all the results in this section.



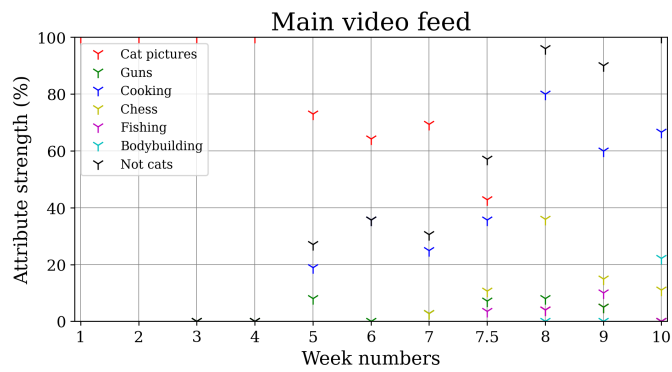
(a) The percentage of posts for each attribute from liked pages in the main feed.



(b) The percentage of recommended, suggested and sponsored posts for each attribute in the main feed from pages not liked by the user nor the BOT .

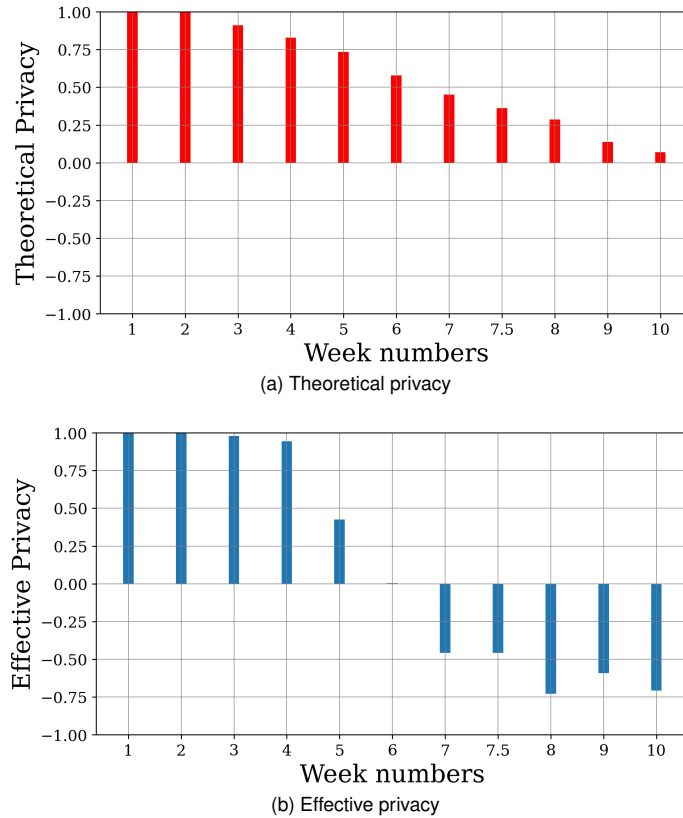


(c) The percentage of video posts for each attribute from the latest video feed.



(d) The percentage of each attribute of video posts from the main video feed.

**Figure 6.2. Effective attribute strength variables with combined noise**



**Figure 6.3.** *Theoretical and Effective privacy results*

## 6.2 Privacy Results

Based on the definitions described in chapter 5, we calculated each week's Theoretical and Effective Privacy values (Figures 6.3a and 6.3b). During the first two weeks, we built the user's real attributes and added increasing noise to render FB's noise feed equal to the real.

Notably, the Effective Privacy in week 6 (50% noise) is close to 0. Once the amount of real traffic generated by users equals the amount of noise traffic, users achieve privacy. The theoretical real attribute strength outweighs the combined noise attribute strengths even after 10 weeks, as shown in 6.1a. This explains the difference between the Theoretical and Effective Privacy values and shows that FB emphasises on the user's recent interests, suggesting a "time of like" variable in its recommendation systems. This also proves that the Effective Privacy is a more accurate way of measuring privacy on a SN.

We added more noise in week 7 and saw a small decrease in the Effective Privacy value – i.e. the account became more private. During week 8, we stopped reinforcing the real attribute to simulate what would happen if the user took a break from FB, while the BOT ran. We noted significant decrease in the Effective Privacy value. Finally, in weeks 9 and 10, we simulated a rarely active user combined with BOT background activity (90% noise). The Effective Privacy value increased as the real attribute was re-enforced again in week

9, while the Effective Privacy value decreased again during week 10.

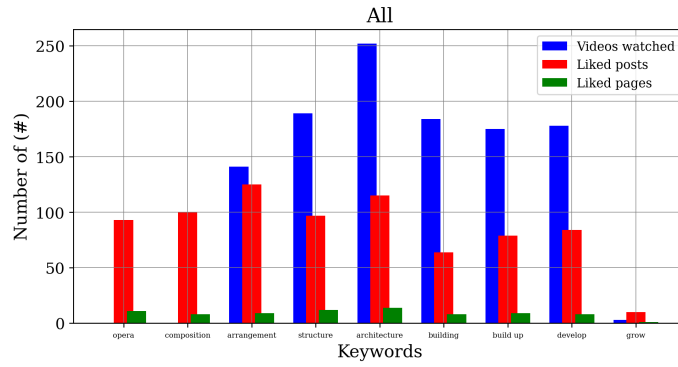
### 6.3 Real Account Results

When evaluating `MetaPriv` extended features on real existing accounts, a different approach was used as compared to that used for the dummy account. The theoretical privacy takes into account all the likes that a user did during their entire history with FB, which is unfeasible to obtain and categorise into keywords. To this end, when analyzing the results, we considered the feed from pages related to the noise keywords as our noise attribute. For the real data, we simply considered the feeds unrelated to our chosen noise keyword. To this end, we ran our experiments for 4 weeks on three existing accounts (account A, B and C). Accounts A, B and C were set to like an average of 27, 54 and 22 posts per day respectively.

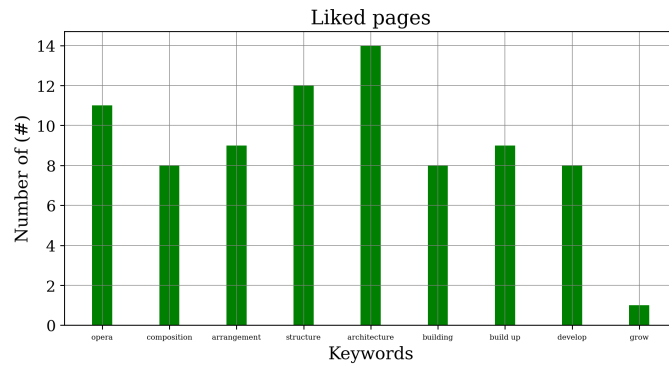
**Account A:** During the 4 week period, the BOT liked 754 posts from 79 pages and watched 1122 videos. The chosen seed keyword was ‘opera’ after which the BOT generated other related noise keywords. These keywords, along with their respective amount of posts, pages and videos can be seen in Figure 6.4. During the first week of our evaluations, the BOT used the first two keywords: ‘opera’ and ‘composition’. We then analyzed the results, and again observed a complete absence of noise from the FB feeds (this eventually led to the development of the video watching and link clicking features). The extended version of `MetaPriv` run for the subsequent 3 weeks.

We then analyzed the different FB feeds. In the main feed, out of 596 posts, 86 were related to real interests and 127 were noise related. From 136 suggested posts, 11 were based on real interests, 67 were based on noise and 58 seemed to be related only to location (local grocery advertisements, etc.). In the latest video feed, from 132 videos, 123 were related to real interests and 9 to noise. Finally, in the main video feed, out of 300 videos, 27 were real interest related, 15 were noise related and the rest seemed unrelated to noise or real interests. These results are represented graphically in Figure 6.5. 6.5a shows the exact number of real and noise data (the unrelated bar is out of bounds as it is not used in the calculations) and 6.5b shows the percentage of real and noise data.

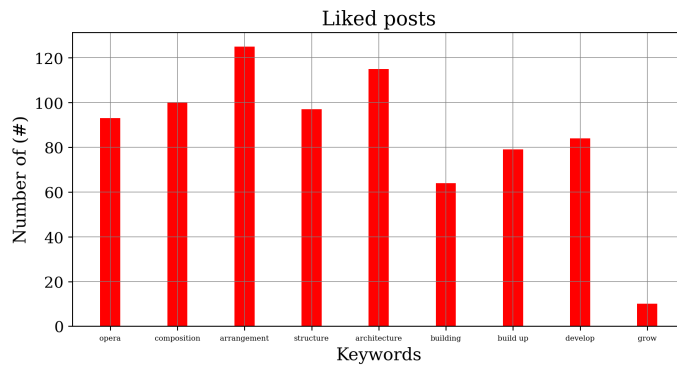
**Account B:** For this account, the BOT liked 1518 posts from 129 pages and watched 2871 videos. The chosen seed keyword was ‘toyota’ and the keyword statistics are shown in Figure 6.6. With this account, the extended `MetaPriv` was used from the beginning. Once completed, we analyzed the different FB feeds. In the main feed, out of 300 posts, 79 were real interest related and 27 were noise related. From 59 suggested posts, 4 were based on real interests, 4 were based on noise and 51 seemed to be related only to location (local grocery advertisements, data carriers, etc.). In the latest video feed, from 300 videos, 100 were related to real interests and 200 were related to noise. Finally, in



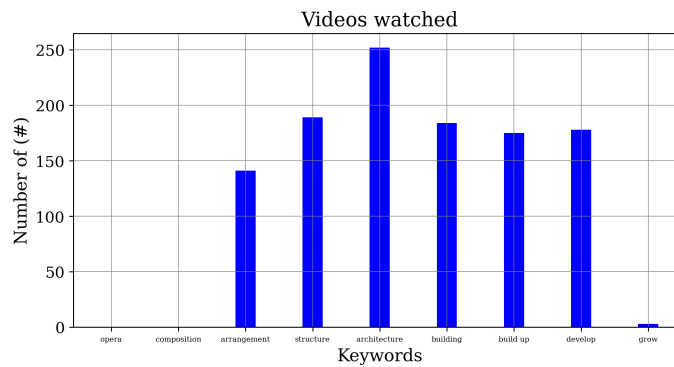
(a) Keywords and their respective amount of liked posts, liked pages and watched videos



(b) Keywords and their respective amount of liked pages

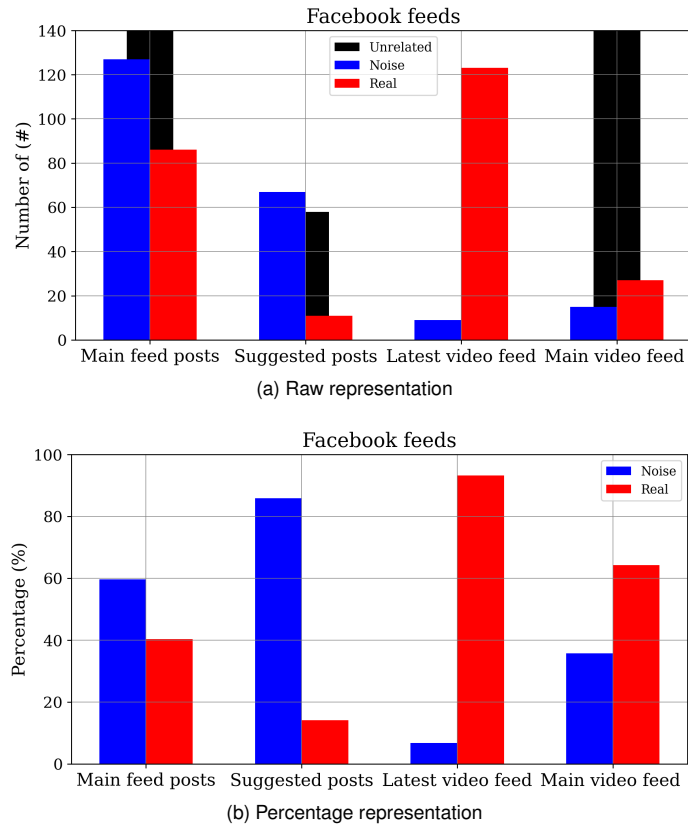


(c) Keywords and their respective amount of liked posts



(d) Keywords and their respective amount of watched videos

**Figure 6.4.** Keyword statistics of account A.

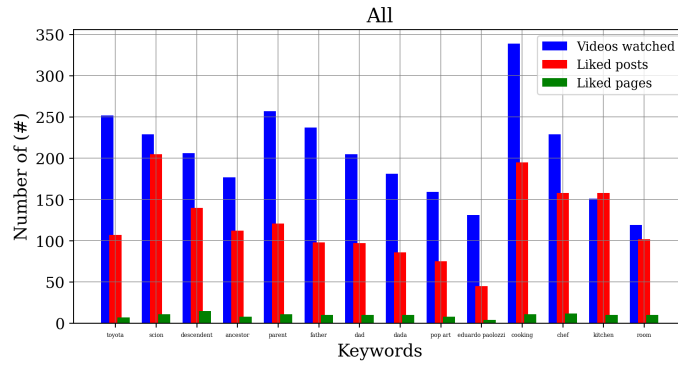


**Figure 6.5.** Graphical representation of the collected data from account A.

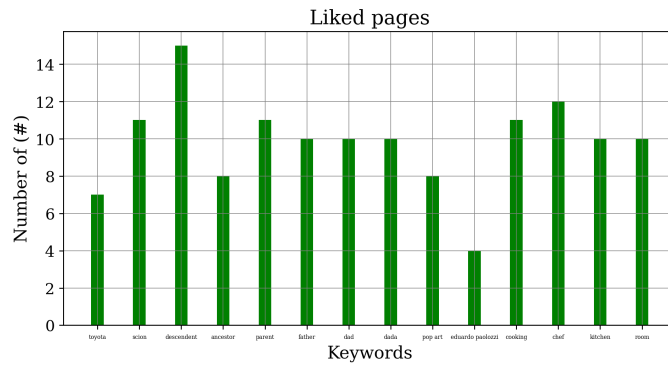
the main video feed, out of 300 videos 30 were real interest related, 14 were noise related and the rest seemed unrelated to noise or real interests. The results are also represented graphically in Figure 6.7.

**Account C:** In the last account, the BOT liked 609 posts from 182 pages and watched 110 videos. With this account, we used 71 different keywords to see how the amount of FB page diversity affects the final results. Moreover, the small amount of video watches also allows us to see how the feeds are affected by watching videos. After analyzing the results, in the main feed, out of 292 posts, 17 were real interest related and 83 were noise related. From 143 suggested posts, 58 were based on real interests, 41 were based on noise and 44 seemed to be related only to location (local grocery advertisements, data carriers, etc.). In the latest video feed, from 275 videos, 169 were related to real interests and 106 were related to noise. Finally, in the main video feed, out of 282 videos, 100 were real interest related, 24 were noise related and the rest seemed unrelated to noise or real interests. The results are also represented graphically in Figure 6.8.

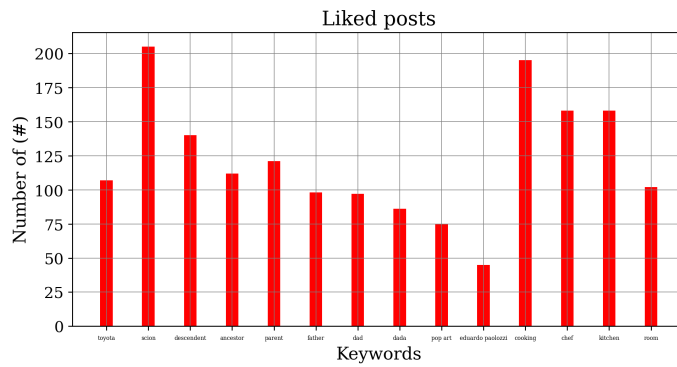
It should be noted that analyzing the feed of a real account can be a tedious process as every post has to be manually inspected. Implementing a script to automate this task would be very challenging as there is no way it could distinguish between posts from pages and posts from friends and groups. Additionally, for the FB feed that came as suggested (suggested posts and main video feed), it was harder to understand what was



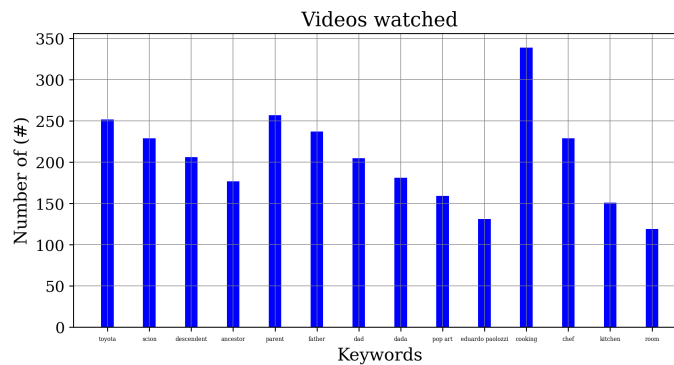
(a) Keywords and their respective amount of liked posts, liked pages and watched videos



(b) Keywords and their respective amount of liked pages

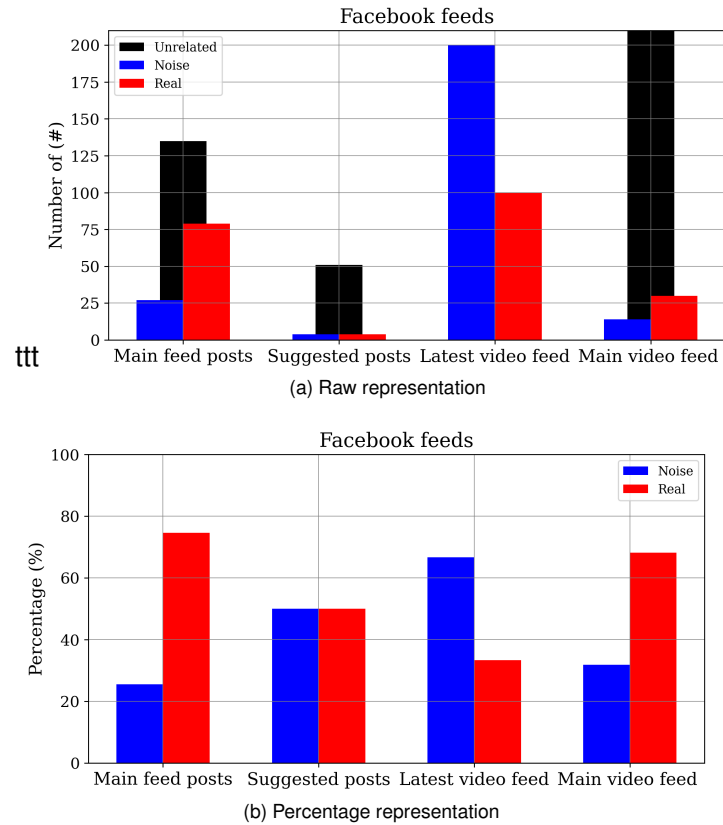


(c) Keywords and their respective amount of liked posts



(d) Keywords and their respective amount of watched videos

**Figure 6.6.** Keyword statistics of account B.

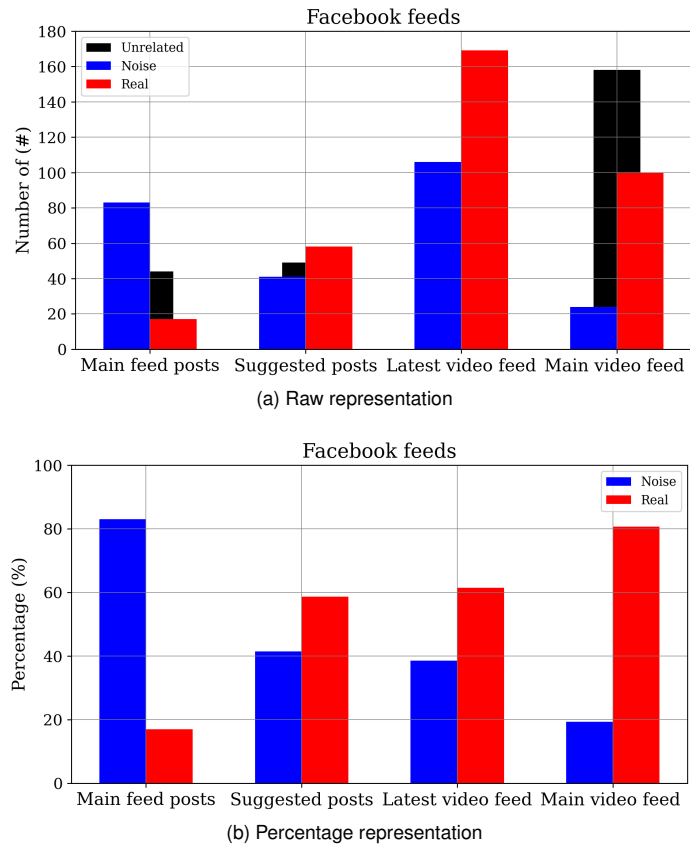


**Figure 6.7.** Graphical representation of the collected data from account B.

related to the real or noise data. With the old accounts there are significantly more variables to consider such as friends, devices, locations etc. as opposed to a new account that exists in a controlled environment. Hence, the results should be treated as an approximation.

**Observations:** We compared the percentage representation of the FB feeds from the 3 accounts: 6.5b, 6.7b and 6.8b.

1. Account A has significantly more noise from suggested posts as compared to account B and C. We believe account A had more keywords that were of interest to FB. When account A searched for keywords like development, building and growth, FB showed pages that generally were more popular and posted a lot on their pages. This, in turn, made resulted in an intense noise pollution with suggested posts which are in essence ads. Since ads are FBs main profit source, we believe that these specific interests likely bring FB more profit as users with these interests can be advertised exact products they can easily buy.
2. Account B has significantly more noise related feed in its latest video feed. Initially, we believed that it is because account B had the keywords cooking, chef and kitchen which produced a lot of video feed about cooking recipes. However, account C also had a lot of cooking related keywords but its latest video feed is not so



**Figure 6.8.** Graphical representation of the collected data from account C.

well polluted with noise.

- Account C has significantly more noise in its main feed as compared to account A and B. This is likely a result of keyword variety. Since account C has significantly more keywords, it is more likely that more keywords lead to pages that post a lot of content on a daily basis. Also, one of the keywords was football and account C ran MetaPriv during the 2022 football world cup. This led to a strong amount of football related noise in the main feed since football can be considered a hot topic at that time.
- Account C has the least amount of noise in its main video feed. This is probably because the main video feed is influenced by what and how many videos a user watches. The small amount of video watches most likely caused this, since this feed mostly includes videos from pages unknown by the user or the BOT (therefore can be considered as suggested videos).

From these observations, we can draw a number of conclusions. The main feed is mainly influenced by hot topics and pages that post a lot of content. Next, the suggested posts are influenced by topics that can be monetized by FB, i.e. topics that result in ads that make FB more money. The latest video feed is only influenced by how much content different pages post per day. Also it is only dependent on page likes and not on video

watches. Lastly, the main video feed is influenced by what and how many videos a user watches.

Nonetheless, with the acquired data, we proceeded to calculate the effective privacy. Posts from friends were ignored since those usually did not reflect any particular attribute of the user. By taking these results into account, the effective privacy yields a result of 0.06 for account A, 0.13 for account B and 0.09 for account C. All the accounts seem to have achieved a larger degree of privacy, with account A and C basically having its noise attributes almost indistinguishable from its real attributes. This, according to chapter 5, means that account A and C has achieved privacy. Account B however, likely needs more time to run the tool.

## 6.4 Limitations

After running experiments with `MetaPriv`, we observed the following limitations in its usability. Firstly, the user's FB feeds become less appealing as they become more and more infested with noise related posts. A solution for this would be to tone down the amount of noise the tool generates. Another option is to use a noise filtering tool we have developed, which is described in section 4.3.

Another limitation is the variety of noise data that a tool can generate automatically. In our implementation, we used liking posts, pages and watching videos as ways to generate noise. These were chosen as they represent direct interest of the topic at hand. Nevertheless, they are only a fraction of things that a normal user can do on FB. Other interactions such as user posting, commenting, reacting to posts and playing games are likely also used by FB for better user profiling. These interactions however are very hard to simulate adequately and can lead to unwanted issues such as generating posts that are unintelligible, inappropriate or even extremist, which, in the end, can lead to account blocking.

It is also worth mentioning that some users may be hesitant to use `MetaPriv` out of fear of liking something inappropriate. This is one of the reasons that the noise generated by the tool is not completely random. Mainly, the user chooses a seed keyword that will define the first noise attribute and start generating traffic based on it. The next noise attribute will be generated based on the initial seed – a word that is related to the seed word. We admit that this might not be a perfect solution and solutions can be further developed in future works. Another concern is generating traffic related to illegal, extremist or abuse topics. This traffic however is constantly removed from the platform.

Finally, in order to run experiments with `MetaPriv` or use it as it is, a user needs to keep the tool constantly running on their device. This becomes quite inconvenient when running the tool on a device such as a laptop (a situation we encountered during our

experiments). To resolve this limitation, we introduce an alternate use case protocol, where `MetaPriv` is deployed in the cloud and communicates to FB on behalf the user.

## 7. USE CASE PROTOCOL AND SECURITY ANALYSIS

In this section, we introduce our alternate use case protocol for MetaPriv and formalize the communication between a user  $u_i$  and the BOT. Additionally, we prove our construction's security against a threat model defined in section 7.2. For this protocol, we consider a scenario, where a user  $u_i$  securely sends a keyword to the BOT deployed in a remote cloud service provider such as the one presented in [29, 30]. We define a keyword chosen by  $u_i$  as an attribute  $att_i$  and define the list of attributes maintained by BOT submitted by  $u_i$  as  $\mathcal{A}_i^n$ .

### 7.1 Use Case Protocol

We assume the existence of an IND-CPA secure symmetric key encryption scheme  $SKE = (\text{Gen}, \text{Enc}, \text{Dec})$ . Moreover, we further assume that  $u_i$  and the BOT communicate over a symmetrically encrypted channel, using a shared symmetric key  $K_{u_iB}$  generated as  $K_{u_iB} \leftarrow SKE.\text{Gen}(1^\lambda)$ , where  $\lambda$  is the security parameter of SKE.

The protocol is initiated by  $u_i$  each time they add a new attribute to the BOT's list  $\mathcal{A}_i^n$ . To do so,  $u_i$  picks an attribute  $att_i$ , encrypts it using  $K_{u_iB}$  as  $c_{att_i} \leftarrow SKE.\text{Enc}(K_{u_iB}, att_i)$  and sends the following message to the BOT:

$$m = \langle t, c_{att_i}, \text{HMAC}(K_{u_iB}, t || c_{att_i}) \rangle,$$

where  $t$  is a timestamp and HMAC is a keyed-hash message authentication code operating as a pseudorandom function (PRF). Upon receiving  $m$ , the BOT verifies the freshness and integrity of the message by checking the timestamp and the HMAC respectively. If any verification fails, the BOT outputs  $\perp$  and aborts the protocol. Otherwise, it stores  $c_{att_i}$  to its list of attributes.

### 7.2 Threat Model

We consider an attacker  $\mathcal{ADV}$  who can act in the following malicious ways:

1. Corrupt the entire SN and break user privacy by learning their interests;

2. Break user privacy by corrupting `MetaPriv` and finding the noisy attributes created for each user;
3. Change the user's noisy data mainly to gain profit or market advantage against competitors (e.g. change noisy data to present targeted advertisements to users).

Based on these three malicious behaviours, we have defined a set of attacks available to  $\mathcal{ADV}$ .

**Definition 3** (Traditional Profiling Attack). *Let  $u_i$  be a legitimate user signed on to a social network platform SN. In addition to that, let  $\mathcal{ADV}$  be a curious adversary that has corrupted SN. Then,  $\mathcal{ADV}$  can successfully launch a Traditional Profiling Attack, if, for each registered user,  $u_i$  gets a detailed and accurate list by SN with the entire activity of  $u_i$  in the platform (e.g. likes, follows, posts, etc.).*

**Definition 4** (Noise Data Substitution Attack). *Let  $\mathcal{ADV}$  be an adversary that overhears the communication between the BOT and a user  $u_i$ . In addition, assume that  $u_i$  wishes to add an attribute  $a_f$  to  $u_i$ 's list  $\mathcal{A}_i^n$ .  $\mathcal{ADV}$  successfully launches a Noise Data Substitution Attack, if they manage to replace the attribute  $a_n$  with another of their choice without the BOT realizing it and eventually adding it to `MetaPriv`'s database for user  $u_i$ .*

**Definition 5** (Noisy Attribute Identification Attack). *Let  $\mathcal{ADV}$  be a malicious adversary, who overhears the communication between the BOT and a legitimate user  $u_i$ . Additionally, assume that  $\mathcal{ADV}$  gains access to the database where  $u_i$ 's data generated by `MetaPriv` is stored. Then,  $\mathcal{ADV}$  launches a successful Noisy Attribute Identification Attack either by intercepting the exchanged messages between  $u_i$  and the BOT or by examining stored user data and correctly finding at least some of the noisy attributes used by `MetaPriv`.*

### 7.3 Security Analysis

Here, we prove the security of our construction against the threat model of section 7.2. We begin this Section with a brief discussion on the *Traditional Profiling Attack* as per Definition 3.

**Traditional Profiling Attack:** To successfully perform a *Traditional Profiling Attack* against a user  $u_i$ , an adversary  $\mathcal{ADV}$  needs a detailed list by the SN containing  $u_i$ 's full activities. However, our extensive experimental control shows that our construction can achieve full privacy after 6 weeks. As discussed in Section 6.2, *effective privacy* is a more accurate index for Social Networks compared to *theoretical privacy*. Hence, we can conclude that a user  $u_i$  can fully prevent a *Traditional Profiling Attack* through our construction after 6 weeks. However, since our construction allows users to **quantify** their privacy, each user can prevent the attack fully or partially or refrain from preventing it.

**Proposition 1** (Noise Data Substitution Attack Soundness). *Let  $\mathcal{ADV}$  be a malicious ad-*

versary overhearing communication between a user  $u_i$  and the BOT. Moreover, let SKE be an IND-CPA secure symmetric-key cryptosystem and HMAC a key-message authentication code, proved to be a PRF. Then  $\mathcal{ADV}$  cannot successfully perform a Noise Data Substitution Attack.

*Proof.*  $\mathcal{ADV}$  will successfully launch a Noise Data Substitution Attack if they tamper with message  $m = \langle t, c_{\text{att}_i}, \text{HMAC}(K_{u_i B}, t || c_{\text{att}_i}) \rangle$  sent by  $u_i$  to the BOT. To do so,  $\mathcal{ADV}$  must satisfy at least one of the following:

**C1:** Replace  $c_{\text{att}_i}$  with another ciphertext  $c_{\mathcal{ADV}}$  encrypting an attribute of their choice;

**C1:** Replay an old message.

- **C1** will hold, if  $\mathcal{ADV}$  (1) picks an attribute  $\text{att}_{\mathcal{ADV}}$  of choice, (2) gets the symmetric key copy  $K_{u_i B}$ , (3) encrypts  $\text{att}_{\mathcal{ADV}}$  using  $K_{u_i B}$ , (4) generates a valid HMAC and (5) swaps message components  $m$  with malicious ones. However, given the IND-CPA security of SKE,  $\mathcal{ADV}$  can only recover the symmetric key with probability negligible in  $\lambda$ , where  $\lambda$  is the security parameter of SKE. Thus,  $\mathcal{ADV}$  can satisfy **C1** only with negligible probability.
- The other option for  $\mathcal{ADV}$  is to replay an older valid message:  $\mathcal{ADV}$  intercepts  $m$  and replaces it with a previously intercepted  $m_{\text{old}}$ . However, since the HMAC portion of the message contains a timestamp,  $\mathcal{ADV}$  would need to create a new valid HMAC with a new timestamp. Similarly to **C1**, this can only occur with knowledge of  $K_{u_i B}$  and hence, with negligible probability.

As a result, both **C1** and **C2** can be satisfied with negligible probability, and thus,  $\mathcal{ADV}$  can launch a successful Noise Data Substitution Attack only with negligible probability.

□

**Proposition 2** (Noisy Attribute Identification Attack Soundness). *Let  $\mathcal{ADV}$  be a malicious adversary overhearing communication between  $u_i$  and the BOT and having access to the BOT's database. Let SKE be an IND-CPA secure symmetric-key cryptosystem. Then  $\mathcal{ADV}$  cannot launch a successful Noisy Attribute Identification Attack.*

*Proof.* Attributes are both transferred, stored and encrypted under  $K_{u_i B}$ . Hence, even if  $\mathcal{ADV}$  intercepts the message  $m$  sent from  $u_i$  to the BOT, access to  $K_{u_i B}$  is required to recover the attribute's value. Similarly, even with access to the BOT's database,  $\mathcal{ADV}$  would still need  $K_{u_i B}$  to decrypt all stored ciphertexts. However, given the IND-CPA security of SKE,  $\mathcal{ADV}$  can only recover  $K_{u_i B}$  with probability negligible in  $\lambda$ . Hence,  $\mathcal{ADV}$  can launch a successful Noisy Attribute Identification Attack only with negligible probability.

□

## 8. CONCLUSION AND SOCIETAL IMPACT

Social networks shaped the digital world becoming an indispensable part of our daily lives. Over the years, these platforms have gained a reputation for tracking user online activity. These strategies may prove threatening for multiple spheres of peoples' lives – spanning from consumption to opinion formation – and may have ominous effects on democracy [31, 32]. This vast collection of personal data by SNs is often exposed (i.e. sold) to third-party companies. Additionally, SN users do not usually have a say on the information they access, as SNs prioritize the content presented on feeds, based on what users most probably want to see. In other words, SN algorithms seemingly hide content and have a great impact on the information users are able to reach. With privacy and societal concerns over SNs rapidly rising, these platforms are seen as rather controversial.

Having identified these issues, we built `MetaPriv`, a tool that adds new privacy safeguards for SN users aimed at hampering SN ability to serve targeted content. `MetaPriv` allows users to define their desired level of privacy, and the idea is based on obfuscation, which in this context means achieving privacy by making Facebook collect noise data from the users' accounts. In this way, `MetaPriv` strikes a balance between privacy and functionality. Moreover, the tool can be further developed to accommodate other social media platforms such as Twitter, Instagram, Youtube etc. The only requirement is a device that can run a web browser since all these platforms have a browser version that can be automated using the web automation tools. We believe this feature will be used in several services in the near future and will help towards building less biased SNs, while minimizing the amount of personal information processed by platforms.

## REFERENCES

- [1] A. Michalas, T. Dimitriou, T. Giannetsos, N. Komninos, and N. Prasad. “Vulnerabilities of Decentralized Additive Reputation Systems Regarding the Privacy of Individual Votes”. English. *Wireless Personal Communications* 66.3 (2012), pp. 559–575. ISSN: 0929-6212.
- [2] A. Michalas and M. Bakopoulos. “SecGOD Google Docs: Now I Feel Safer!”: *2012 International Conference for Internet Technology And Secured Transactions*. Dec. 2012, pp. 589–595.
- [3] T. Dimitriou and A. Michalas. “Multi-Party Trust Computation in Decentralized Environments”. *2012 5th International Conference on New Technologies, Mobility and Security (NTMS)*. May 2012, pp. 1–5. DOI: 10.1109/NTMS.2012.6208686.
- [4] T. Dimitriou and A. Michalas. “Multi-party Trust Computation in Decentralized Environments in the Presence of Malicious Adversaries”. *Ad Hoc Networks* 15 (Apr. 2014), pp. 53–66. ISSN: 1570-8705. DOI: 10.1016/j.adhoc.2013.04.013. URL: <http://dx.doi.org/10.1016/j.adhoc.2013.04.013>.
- [5] Meta. *The Facebook pixel – measure, optimise and build audiences for your advertising campaigns*. <https://www.facebook.com/business/learn/facebook-ads-pixel>. [Online; accessed 7-July-2022]. 2022.
- [6] W. Luo, Q. Xie, and U. Hengartner. “Facecloak: An architecture for user privacy on social networking sites”. In *Proceedings of 2009 IEEE International Conference on Privacy, Security, Risk and Trust (PASSAT-09)*. 2009, p. 1.
- [7] A. Michalas. “The Lord of the Shares: Combining Attribute-based Encryption and Searchable Encryption for Flexible Data Sharing”. *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*. SAC '19. Limassol, Cyprus: ACM, 2019, pp. 146–155. ISBN: 978-1-4503-5933-7. DOI: 10.1145/3297280.3297297. URL: <http://doi.acm.org/10.1145/3297280.3297297>.
- [8] Q. X. Wanying Luo and U. Hengartner. *FaceCloak implementation download*. <https://crisp.uwaterloo.ca/software/facecloak/download.html>. [Online; accessed 7-July-2022]. 2010.
- [9] F. Beato, M. Kohlweiss, and K. Wouters. “Scramble! Your Social Network Data”. *Privacy Enhancing Technologies*. Ed. by S. Fischer-Hübner and N. Hopper. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 211–225.
- [10] L. Hull. *Hey advertisers, track THIS*. <https://blog.mozilla.org/products/firefox/hey-advertisers-track-this/>. [Online; accessed 7-July-2022]. 2019.

- [11] X. Xing, W. Meng, D. Doozan, A. C. Snoeren, N. Feamster, and W. Lee. "Take This Personally: Pollution Attacks on Personalized Services". *22nd USENIX Security Symposium (USENIX Security 13)*. Washington, D.C.: USENIX Association, Aug. 2013, pp. 671–686. ISBN: 978-1-931971-03-4. URL: <https://www.usenix.org/conference/usenixsecurity13/technical-sessions/paper/xing>.
- [12] W. Meng, X. Xing, A. Sheth, U. Weinsberg, and W. Lee. "Your Online Interests: Pwned! A Pollution Attack Against Targeted Advertising". *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. CCS '14*. Scottsdale, Arizona, USA: Association for Computing Machinery, 2014, pp. 129–140. ISBN: 9781450329576. DOI: 10.1145/2660267.2687258. URL: <https://doi.org/10.1145/2660267.2687258>.
- [13] I. L. Kim, W. Wang, Y. Kwon, Y. Zheng, Y. Aafer, W. Meng, and X. Zhang. "Adbudgetkiller: Online advertising budget draining attack". *Proceedings of the 2018 World Wide Web Conference*. 2018, pp. 297–307.
- [14] N. Vincent, B. Hecht, and S. Sen. "'Data Strikes': Evaluating the Effectiveness of a New Form of Collective Action Against Technology Companies". *The World Wide Web Conference. WWW'19*. San Francisco, CA, USA: ACM, 2019, pp. 1931–1943. ISBN: 9781450366748.
- [15] I. Arrieta-Ibarra, L. Goff, D. Jiménez-Hernández, J. Lanier, and E. G. Weyl. "Should We Treat Data as Labor? Moving beyond 'Free'". *AEA Papers and Proceedings* (2018). ISSN: 25740768, 25740776.
- [16] D. C. Howe and H. Nissenbaum. "Engineering privacy and protest: A case study of AdNauseam". English (US). *CEUR Workshop Proceedings 1873* (2017), pp. 57–64. ISSN: 1613-0073.
- [17] J. Zhang, K. Psounis, M. Haroon, and Z. Shafiq. "HARPO: Learning to Subvert Online Behavioral Advertising". *arXiv preprint arXiv:2111.05792* (2021).
- [18] Meta. *Facebook Reports First Quarter 2021 Results*. <https://investor.fb.com/investor-news/press-release-details/2021/Facebook-Reports-First-Quarter-2021-Results/default.aspx>. [Online; accessed 31-December-2022]. 2021.
- [19] Meta. *Suspending Cambridge Analytica and SCL Group From Facebook*. <https://about.fb.com/news/2018/03/suspending-cambridge-analytica/>. [Online; accessed 10-July-2022]. 2018.
- [20] J. Constine. *Facebook pays teens to install VPN that spies on them*. <https://techcrunch.com/2019/01/29/facebook-project-atlas/>. [Online; accessed 10-July-2022]. 2019.
- [21] H. Kelly. *Facebook bug set 14 million users' sharing settings to public*. <https://money.cnn.com/2018/06/07/technology/facebook-public-post-error/index.html>. [Online; accessed 10-July-2022]. 2018.
- [22] Y. Angel. *obfs4 - The obfourscator*. <https://github.com/Yawning/obfs4/blob/master/doc/obfs4-spec.txt>. [Online; accessed 31-December-2022]. 2022.

- [23] F. Brunton and H. Nissenbaum. *Obfuscation: A user's guide for privacy and protest*. Mit Press, 2015.
- [24] Mozilla. *WebDriver*. <https://developer.mozilla.org/en-US/docs/Web/WebDriver>. [Online; accessed 31-December-2022]. 2022.
- [25] E. Aghasian, S. Garg, and J. Montgomery. *User's Privacy in Recommendation Systems Applying Online Social Network Data, A Survey and Taxonomy*. 2018.
- [26] E. M. Maximilien, T. Grandison, K. Liu, T. Sun, D. Richardson, and S. Guo. "Enabling privacy as a fundamental construct for social networks". *2009 International Conference on Computational Science and Engineering*. Vol. 4. IEEE. 2009.
- [27] J. Domingo-Ferrer. "Rational privacy disclosure in social networks". *International conference on modeling decisions for artificial intelligence*. Springer. 2010.
- [28] R. Cantaragiu, A. Michalas, E. Frimpong, and A. Bakas. "MetaPriv: Acting in Favor of Privacy on Social Media Platforms". *Security and Privacy in Communication Networks*. Ed. by F. Li, K. Liang, Z. Lin, and S. K. Katsikas. Cham: Springer Nature Switzerland, 2023, pp. 692–709. ISBN: 978-3-031-25538-0.
- [29] N. Paladi, C. Gehrman, and A. Michalas. "Providing User Security Guarantees in Public Infrastructure Clouds". *IEEE Transactions on Cloud Computing* 5.3 (July 2017), pp. 405–419. DOI: 10.1109/TCC.2016.2525991.
- [30] N. Paladi, A. Michalas, and C. Gehrman. "Domain Based Storage Protection with Secure Access Control for the Cloud". *Proceedings of the 2014 International Workshop on Security in Cloud Computing*. ASIACCS '14. Kyoto, Japan: ACM, 2014. ISBN: 978-1-4503-2805-0.
- [31] T. Khan, A. Michalas, and A. Akhunzada. "Fake news outbreak 2021: Can we stop the viral spread?": *Journal of Network and Computer Applications* 190 (2021), p. 103112. ISSN: 1084-8045.
- [32] T. Khan and A. Michalas. "Trust and Believe - Should We? Evaluating the Trustworthiness of Twitter Users". *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. 2020, pp. 1791–1800. DOI: 10.1109/TrustCom50675.2020.00246.