Automatic de-identification of Data Download Packages

Laura Boeschoten [a,*], Roos Voorvaart [b] and Ruben Van Den Goorbergh [c] Casper Kaandorp [d] Martine De Vos [e]

[a] *Department of Methodology and Statistics Utrecht University, The Netherlands*
E-mail: l.boeschoten@uu.nl; ORCID: https://orcid.org/0000-0002-3536-0474
[b] *Research and Data Management Services, Utrecht University, The Netherlands*
E-mail: r.voorvaart@uu.nl; ORCID: https://orcid.org/0000-0002-4411-8495
[c] *Department of Methodology and Statistics Utrecht University, The Netherlands*
E-mail: r.vandengoorbergh@uu.nl; ORCID: https://orcid.org/0000-0003-3229-3015.
[d] *Research and Data Management Services, Utrecht University, The Netherlands*
E-mail: c.s.kaandorp@uu.nl; ORCID: https://orcid.org/0000-0001-6326-6680
[e] *Research and Data Management Services, Utrecht University, The Netherlands*
E-mail: m.g.devos@uu.nl; ORCID: https://orcid.org/0000-0001-5301-1713

Abstract. The General Data Protection Regulation (GDPR) grants all natural persons the right ~~of access to~~to access their personal data if this is being processed by data controllers. The data controllers are obliged to share the data in an electronic format and often provide the data in a so called Data Download Package (DDP). These DDPs contain all data collected by public and private entities during the course of a citizens' digital life and form a treasure trove for social scientists. However, the data can be deeply private. To protect the privacy of research participants while using their DDPs for scientific research, we developed a de-identification ~~software~~algorithm that is able to handle typical characteristics of ~~DDPs such as regularly changing file structures, visual and textual content, different file formats, different file structures and accounting for usernames.~~DDPs. These include regularly changing file structures, visual and textual content, differing file formats, differing file structures and private information like usernames. We investigate the performance of the ~~software~~algorithm and illustrate how the ~~software~~algorithm can be tailored towards specific DDP structures.

Keywords: Data Download Package, Instagram, De-identification, Anonymization, Pseudonymization

1. Introduction

~~The General Data Protection Regulation (GDPR) grants all natural persons the right of access to their personal data if this is being processed by data controllers, such as tech companies, governments and mobile phone providers [1]. Data controllers are obliged to provide a copy of this personal data in a machine readable format and most large data controllers currently comply with this by providing users with the option to retrieve an electronic "Data Download Package" (DDP). These DDPs contain all data collected by public and private entities during the course of citizens' digital life and form a new treasure trove for social scientists [2, 3]. However, depending on which data controller is used, the data collected~~

[*] Corresponding author. E-mail: l.boeschoten@uu.nl.

~~through DDPs can be deeply private and potentially sensitive. Therefore, collecting DDPs for scientific research raises serious privacy concerns and it would not be in line with the principles listed in the GDPR if appropriate measures to protect the privacy of research participants donating their DDPs would not be taken.~~

~~To protect privacy of research participants while using DDPs for scientific research, different types of security measures should be taken such as using shielded (cloud)environments to store the data and using privacy-preserving algorithms when analyzing the data. One key issue here is that the privacy of the participants should be preserved while their data is investigated by researchers and that, although appropriate security measures are taken to prevent this, in case of a data breach, it should not be possible to identify research participants. Because of these reasons, a thorough de-identification procedure is imperative. Many different types of software are already available for this, such as DEDUCE [4] and 'de-identify Twitter' [5]. However, existing methods are not able to handle the highly complex and unstructured nature of DDPs. A particular characteristic of DDPs, that a de-identification procedure should consider, is the fact that the primary identifier of a natural person can be different for different DDPs and is often a username. Furthermore, some DDPs store private interactions of research participants with their contacts, which should be de-identified as well. At last, in case of personal data protected by the GDPR, 'machine readable' unfortunately does not mean equally structured nor easy to parse. Due to this great variety in content and structure, a new method for de-identification of DDPs is essential.~~

~~In this research project we developed an automatic de-identification approach that can deal with the variety in DDPs. In the development we focused on DDPs from Instagram but we believe that our approach forms the basis of the de-identification of most DDPs and can easily be extended in order to de-identify DDPs from other companies.~~

Although the European Union (EU)s General Data Protection Regulation (GDPR) is often known for restricting the possibilities for owners of databases (“data controllers"), Article 15 of the GDPR unexpectedly also provides many opportunities for data analysts [1]. This article grants data subjects the right to receive a copy of all their personal data collected by a data controller in a machine-readable electronic format. Most data controllers currently comply with this article by providing a so called “Data Download Package" (DDP) to the data subjects upon request. The GDPR also grants the data subject the right to share the DDP with third parties, such as researchers. As these DDPs represent the unique digital fingerprint of individuals who use digital platforms, ranging from bank transactions and purchase history to social media interactions and location history, DDPs form a (still undiscovered) treasure trove for research [2].

However, the data present in DDPs can be deeply private and potentially sensitive. This poses a major challenge to using DDPs for scientific research. Participants might not be willing to share this sensitive data. However, researchers are often only interested in a part of the DDP and do not need the sensitive data. Although an interesting solution is to extract relevant features locally on the device of the participant [3], this workflow is not suitable for all research purposes. When, for example, an exploratory approach is of interest, or when the aim is to develop or improve the performance of an extraction algorithm, local extraction would limit the analytic possibilities. In such situations, collection of the *complete* DDP is desired, which requires challenges caused by the sensitivity of the data to be overcome. An example of such a research project is Project AWeSome [6], which collects complete Instagram DDPs from research participants. The participants' DDPs are stored in a secured environment where they are de-identified using the de-identification algorithm proposed in this manuscript. Only after the sensitive information is adequately masked, can the DDPs be shared with the applied researchers for substantive analyses.

We argue that in situations where complete DDPs are collected for research, the DDPs should be treated in a similar fashion as any other sensitive data that is collected for research purposes. We therefore follow the guidelines for sensitive research data[1], which were established by Utrecht University for handling sensitive data from official statistical agencies and governmental bodies like Statistics Netherlands [7] and the European Commission[8]. From these guidelines it can be concluded that two important measures should be taken. First, security measures such as using shielded (cloud) environments for data storage should be used. Second, the privacy of participants should be preserved while their data is analysed by researchers.

An automatic de-identification approach is required since a manual approach would by definition violate the privacy of participants. Besides, a manual approach would be prone to errors and too labor intensive due to the potential size of the DDPs. Many different approaches to automatically de-identify data have been developed over the past years for medical documents [e.g., 4, 9–11], twitter data [e.g., 12, 13] and relational or tabular data [e.g., 14, 15]. De-identification of DDPs poses a challenge because the structure and content of DDPs deviate from the structure and content of the data for which these methods were developed. In addition, DDPs show a wide variety and are collected for different research purposes. In this paper we propose an automatic de-identification algorithm that can handle the structure and content of DDPs and is able to deal with the large variability.

Our contributions are the following:

- We give insight in the structure and content of DDPs in general and Instagram DDPs in particular.
- We develop an open source de-identification algorithm ~~and provide it open source.~~
- We create an open source evaluation data ~~set~~corpus ~~and provide it open source~~.
- ~~We prove that our algorithm is able to find and de-identify a substantive amount of personal data within DDPs.~~We provide statistics that illustrate the performance of the developed de-identification algorithm.
- We provide ~~the~~ open source validation algorithm and ground truth ~~used open source~~.

~~In the Background section we describe in more detail the structure of DDPs and we discuss how privacy of research subjects can be preserved when their DDPs are used for scientific research. In the Methods section we describe our de-identification strategy and how we deal with variety in Instagram DDPs. In addition, this section contains a description of the algorithm that we developed. In the Evaluation section we describe the creation of the evaluation data set. In the Results section we describe the outcomes of this evaluation procedure.~~In Section 2, we illustrate how DDPs from different platforms can vary greatly in structure and content. In Section 4, we discuss the current state-of-the-art in terms of de-identification methods and illustrate why these current methods do not suffice for our aim. In Section 3 we describe the data used for the development and evaluation of our proposed algorithm, which is extensively discussed in Section **??**. The outcome of the evaluation study is presented in Section **??**, followed by suggestions for future work in Section 7 and conclusions in Section 8.

[1] https://www.uu.nl/en/research/research-data-management/faq

2. ~~Background~~Data download packages

~~The aim of the software introduced in this paper is to enable researchers to use DDPs for scientific research while preserving the privacy of participants. In this section, we explain in more detail the specific type of data that can be found in DDPs, define our aims in terms of data protection in more detail and discuss relevant existing literature and software.~~

2.1. ~~*Data Download Packages*~~

~~Most large data controllers currently comply with the right of data access by providing users with the option to retrieve an electronic "Data Download Package" (DDP). This DDP typically comes as a .zip-file containing .json, .html, .csv, .txt, .JPEG and/or .MP4 files in which all the digital traces left behind by the data subject with respect to the data controller are stored. The structure and content of a DDP varies per data controller, and even within data controllers there are differences among data subjects. Data subjects may use different features provided by the data controller and this is reflected by their DDP, for example, if a data subject does not share photos on Facebook, there will be no data folder with .JPEG files in the corresponding DDP.~~

~~One particular characteristic of DDPs is that their content and structure is often subject to change. For example, if a data subject downloads the DDP at a data controller, and repeats this a month later, differences may be found in the structure of the DDP. This can have several causes. The most straightforward cause is that the data subject generated additional data throughout this month. However, other important factors also play a role. First, data controllers can develop new features by which new types of data regarding the data subject are collected. Second, other features are phased out. Third, some data (for example search history) is only saved for a limited amount of time and is destroyed by the data controller after that period. In that case, it will also not be present in the DDP anymore. At last, the GDPR is still relatively new and data controllers continue to optimize the processes used to transfer the relevant data to its subjects, leading to changes in the structure of DDPs.~~

2.2. ~~*Instagram DDPs*~~

~~As the software in this research project was initially developed to de-identify Instagram DDPs, the structure of these DDPs has been thoroughly investigated. Instagram DDPs come as one or multiple zipfiles (depending on the amount of data available on the data subject). The .zip-file contains a number of folders in which all the visual content is stored, namely "photos", "videos", "profile" and "stories". The different folders refer to the different Instagram features used by the data subject to generate the visual content. For example, in the folder "profile", a subject's profile picture can be found, while in the folder "stories", visual content can be found generated using the "stories" feature in Instagram, a form of ephemeral sharing. All textual information is collected in a number of .json files. Some of these files have a simple list structure. For example the file "likes.json" lists all the 'likes' given by the subject, supplemented with a timestamp and the username of the Instagram account to which the 'like' was given. Files such as 'connections.json', 'searches.json' and 'seen_content.json' have similar structures. Other files, such as 'profile.json' are typically shorter in size but have a more complex structure, as they typically contain different auxiliary characteristics. Other files with such a structure are for example 'account_history.json', 'devices.json' and 'settings.json'. However, a substantial number of files contains data that is less structured. Examples of such files are 'comments.json', 'media.json', 'messages.json' and 'stories_activities.json'. Furthermore, data subjects at Instagram are not necessarily natural persons.~~

~~Data subjects at Instagram can be identified by a single and unique Username. Typically, natural persons have individual accounts with an accompanying username, but other institutions, such as for example retail shops or bands can also have an individual account with an accompanying username.~~

Most large data controllers currently comply with the right of data access by providing users with the option to retrieve an electronic DDP. This DDP typically comes as a compressed folder containing text and/or media files in which all the digital traces left behind by the data subject with respect to the data controller are stored. Table 1 shows that the content and structure of DDPs differs among data controllers. Differences between DDPs from the same data controller can also occur among data subjects and over time. These differences may be caused by data subjects using different features provided by the data controller or by the fact that the DDP is a snapshot of the data collected by the data subject up to that point. However, other important factors also play a role. First, data controllers can develop new features through which new types of data of the data subject are collected. Second, other features may be phased out. Third, some data (for example search history) is only saved for a limited amount of time and is destroyed by the data controller after that period. In that case, it will also not be present in the DDP anymore. Finally, the GDPR is still relatively new and data controllers continue to optimize the processes used to transfer the relevant data to its subjects, leading to changes in the structure of DDPs.

From Table 1 it can be concluded that the Instagram DDP contains many features that can also be found in DDPs of other data controllers. Common features are the presence of both text and/or media files, the presence of both structured and unstructured text and the presence of specific types of person identifying information (PII). Therefore, an algorithm that is able to de-identify Instagram DDPs also contains the features needed to de-identify many of the DDPs of other data controllers. To summarize, the developed algorithm is able to handle: ~~To summarize, software to de-identify Instagram DDPs should be able to handle:~~

- An ever changing file structure,
- both visual and textual content,
- different file formats,
- ~~Files in highly structured and highly unstructured format and different variants in between~~files ranging from highly structured to highly unstructured formats,
- ~~Natural persons and other users which are identified by their unique username.~~the masking of usernames of natural persons or other users.

2.3. ~~*Presevering privacy of research subjects*~~

~~If DDPs are collected for research purposes, researchers are also considered data controllers and the GDPR applies to them as well [16, p.95]. Among other things, they are obliged to take technical and organisational security measures aiming to minimise the risk of data abuse [16, p.112].~~

~~To determine what type of security measures are exactly appropriate in a situation where DDPs are collected for scientific research, the content of the DDPs and the purpose of the research play an important role. DDPs can contain various types of data. It can be structured or unstructured and can come in many different types of formats. Each researcher can be interested in a different aspects of the DDPs, depending on their research question. One researcher might be interested in the frequency of social media use during a Covid-19 lockdown [17], and uses Instagram DDPs to investigate this. Another researcher might be interested political opinion and electoral success [18] [19] and uses Twitter DDPs. A third researcher might be interested in personality profiling using Facebook “likes" [20].~~

		FACEBOOK[a]	WHATSAPP[b]	TWITTER[c]	SNAPCHAT[d]	INSTAGRAM[e]
DDP INFO	DDP name	facebook-<profile_name>	My account information.zip WhatsApp chat-<group or contactname>.zip	Archive	mydata~<hashed_code>	username_<date_of_download>
	DDP format	.zip	.zip	.zip	.zip	.zip
	Type of files	media, text	media, text	media, text	text	media, text
	Structure	Content folders > content files	Separate DDP per conversation	Content folders < content files	Index file & Format (i.e., json and html) folders > content files	Content text files and Content folders > content media files
MEDIA FILES	Format of media files	.PNG, .JPG, .MP4	.JPG, .MP4, .HTML	.PNG	-	.JPG, .MP4
	Folder structure	All images, videos, stickers are categorized and stored in corresponding folders. There are no loose files.	All images, videos, stickers in single folder	-	All images and videos are categorized and stored in corresponding folders. There are no loose files.	
PII IN MEDIA	Faces	-/+	-/+	-/+	-	-/+
	Written text	-/+	-/+	-/+	-	-/+
	(user)name tags	-	-	-	-	-/+
TEXT FILES	Format of text files	.JSON or .HTML	.TXT, .OPUS, .HTML	.JS or .HTML	.JSON and .HTML	.JSON
	Folder structure	All text files are categorized and stored in corresponding folders. There are no loose files.	All text in single file per conversation	There is one text file per month.	Text files are not categorized and stored in (sub) folders. They are displayed as loose files.	Text files are not categorized and stored in (sub) folders. They are displayed as loose files.
	Structured data	+	+	+	+	+
	Unstructured data (i.e., containing free text)	+	+	+	-/+	+
PII IN TEXT	Usernames	-/+	-/+	+	+	+
	(first) Names	+	+	-/+	-/+	+
	Email addresses	+	+	+	-/+	+
	Phone numbers	-/+	+	-/+	-/+	-/+
	Locations	-/+	-/+	-/+	-/+	-/+

Table 1: Overview of content and structure of DDPs of five data controllers. Note that if a certain object is present in DDPs, this is indicated with +. If it often occurs within the DDP, a -/+ is used. Finally, if said object is not present, a - is used.

[a] https://www.facebook.com/help/1701730696756992
[b] https://faq.whatsapp.com/general/account-and-profile/how-to-request-your-account-information/
[c] https://help.twitter.com/nl/managing-your-account/how-to-download-your-twitter-archive
[d] https://support.snapchat.com/en-US/a/download-my-data
[e] https://help.instagram.com/181231772500920?helpref

~~As can be seen from these examples, some researchers are interested in text, while others are interested in likes or visual content. Consider the situation of a researcher interested in extracting measures of political opinions from text found in DDPs in more detail. Although political opinion is considered a category of sensitive personal data [16, p.79], they are allowed to be collected when necessary for scientific research purposes [16, p.85]. However, as discussed, the researcher collecting this data is obliged to take appropriate security measures such as incorporating data protection measures by design and by default.~~

~~Although the sensitive personal data is typically essential for the researcher, this is not necessarily the information from which identification of research subjects can occur. Research subject identification from a DDP in case of a data breach is much more likely to occur due to the direct personal data that can be found within a DDP. However, direct personal data is less likely to be relevant for the research. Therefore, incorporating a step to remove direct personal data from DDPs in the data processing phase when collecting DDPs for research purposes reduces the probability that a research subject is identified in case of a data breach while it will not affect the quality of the data needed to answer the research question.~~

2.4. Related work

~~To remove direct personal data from DDPs, the software should be able to adhere to the five key characteristics of DDPs introduced in the previous subsection. A first step is to investigate to what extent existing software and literature is able to remove direct personal data from DDPs. A well-known approach is $k-$anonymity [21] which requires that each record in a data-set is similar to at least $k - 1$ other records on the potentially identifying variables [22]. However, parts of the DDPs are highly unstructured and thereby unique per DDP and reaching $k-$anonymity is therefore not feasible. Much research has focused on the de-identification of electronic health records, for example to enable their use in multi-center research studies [23]. Scientific open source de-identification tools are available such as DEDUCE [4] as well as commercial tools, such as Amazon Comprehend [10] and CliniDeID [11] [24]. Similar initiatives have taken place to de-identify personal data in other types of data, such as for human resource purposes [25]. However, textual content generated from structured data-bases such as for electronic health records or human resources typically have a higher level of structure compared to DDPs and does not handle key identifying information in DDPs, such as usernames or visual content and therefore existing software was not sufficient for our purpose. Alternatively, software has been developed focusing on the removal of usernames, for example for Twitter data [5]. Furthermore, many different types of both open source and commercial software are available to identify and blur faces on images and videos, such Microsoft Azure [26], and Facenet-PyTorch [27]. However, none of the investigated software was able to handle both textual and visual content and both structured and unstructured data within one procedure.~~

~~To summarize, a de-identification procedure is required that works appropriately when file structures change rapidly over time, while there are substantive differences in the level of structure within the files, that is able to handle different file formats, that is able to handle both visual and textual content and that recognizes the username as the primary identifier for natural persons, while other types of person identifying information (PII) should also be accounted for, such as first names, phone numbers and e-mail addresses. The developed software aims for such a level of protection that the privacy of the DDP owners (the participants) is always preserved. Importantly, the goal is not to prepare the DDPs for public sharing, however, in the unlikely event of a data breach, the individual research participants~~

~~should not be directly identifiable. Therefore, the de-identification procedure introduced here should always be supplemented with other security measures such as using a shielded (cloud)environment to store the data and using privacy-preserving algorithms when analyzing the data.~~

De-identification of data in the medical domain has extensively been researched. Medical patient data, like electronic health records and clinical notes, are increasingly used for clinical research. As imposed by privacy legislations such as the US Health Insurance Portability and Accountability Act (HIPAA) [28] and the GDPR, the privacy of patients includede in these data has to be protected. Medical data are therefore de-identified by removing all categories of protected health information (PHI) that are defined by the HIPAA. PHI types typically found in medical data are person names and initials, names of institutions, social security numbers and dates [4, 9, 23, 29]. Automatic de-identification approaches in the literature are either rule-based, machine learning based or a combination of both, where machine-learning approaches show the best performance [9, 23, 29]. Scientific open-source de-identification tools are available such as DEDUCE [4] and Amnesia [30] as well as commercial tools, such as Amazon Comprehend [10] and CliniDeID [11] [24]. Most automatic de-identification approaches are constrained to English medical documents and little is known about their generalizability across languages or domains. Although neural networks have shown good generalization performance compared to rule-based and feature based approaches, a substantial decrease of performance has to be expected when applying these out of the box to new languages or domains [9].

User privacy in social media is an emerging research area and has attracted increasing attention recently. To avoid privacy attacks, like identity disclosure and attribute disclosure, publishers of social media data are obliged to protect users' privacy by anonimizing these data before they are published publicly [31]. Anonymizing social media data is a challenging task due to their heterogeneous, highly unstructured and noisy nature [31]. Commonly used statistical disclosure control approaches [14, 15, 21, 22, 30] are designed for relational and tabular data and cannot be directly applied to social media data. In addition, PHI types that are common in medical data are unlikely to be found in textual social media data. These data rather contain person names, usernames or IDs, email addresses and locations [12, 13], but in fact there is limited work on the types of person identifying information (PII) that may be present in textual social media data and how these should be removed [13, 32]. Yet, removing such information has been shown to be far from sufficient in preserving privacy since users' identity or attributes may be inferred from the public data available on social media platforms [31, 33–35]. Finally, social media data may also consist of visual content. Many different types of both open source and commercial software are available to identify and blur faces on images and videos, such as Microsoft Azure [26], and Facenet-PyTorch [27]. However, modern image recognition methods based on deep learning have demonstrated that hidden information in blurred images can be recovered [36].

Like social media data, DDPs are heterogeneous and unstructured and are likely to contain the same types of sensitive information. Yet, the limited de-identification approaches that are available for social media data focus either on textual or visual content and the presence of both types of information within one DDP poses a major de-identification challenge [37]. An important difference is that on social media platforms information on large groups of users is widely available, whereas DDPs are only available for a single individual. The goal of this research is not to prepare the DDPs for public sharing. DDPs will either be stored on the owner's device or in a shielded (cloud)environment and analyzed using privacy-preserving algorithms. In that sense, handling DDP's is comparable to handling medical data and we therefore assume that the risk of privacy attacks is very low. However, for ethical reasons and in the unlikely event of a data breach, DDPs should still be de-identified.

	Information	Instagram DDP
Overall	Main language	Dutch; English
Text	Structure	Unstructured; Loose text files
	Number of files	20
	File names	account_history; autofill; comments; connections; devices; events; fundraisers; guides; information_about_you; likes; media; messages; profile; saved; searches; seen_content; settings; shopping; stories_activities; uploaded_contacts;
	File format	.JSON
Media	Structure	Structured: Folder > subfolder > media files
	Folders	photos; profile; stories; videos
	Subfolders	Date (format: YYYYMM)
	File format	.JPG/.MP4

Table 2

The content of a typical Instagram DDP of a Dutch user

To summarize, we need a de-identification procedure that is able to handle unstructured and heterogeneous data, and can de-identify both visual and textual content within one procedure. It should be able to recognize usernames as the primary identifier for natural persons, while other types of PII, such as person names, phone numbers and e-mail addresses, should also be accounted for.

3. Data

3.1. *Development set*

For the development of this new de-dentification procedure, the researchers initially used two DDPs of their own personal Instagram accounts. The functionality of the algorithm was based on the typical Instagram DDP file structure (see Table 2). To ensure that the developed algorithm can adequately handle possible varieties in DDP structures (over different Instagram accounts), a validation data corpus was created. Using this corpus, the de-identification procedure could be tested and improved, maximizing its effectiveness.

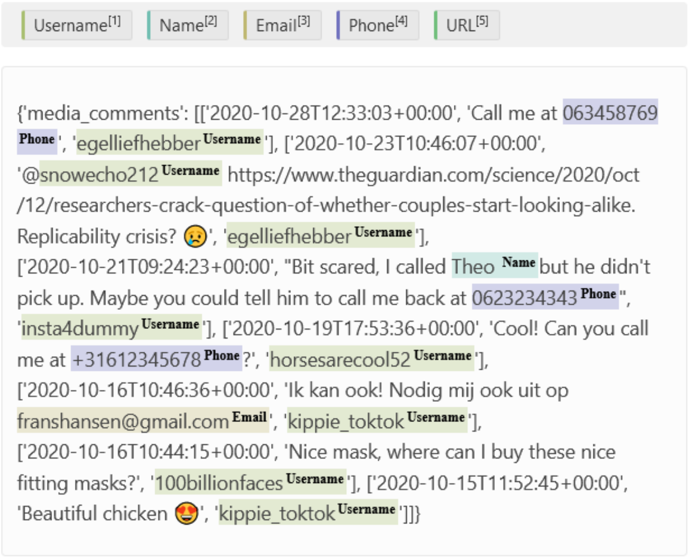
3.2. *Validation corpus sampling*

A group of 11 participants generated Instagram DDPs by actively using a new Instagram account for approximately a week. The participants were instructed not to share any of their own personal information via the Instagram accounts. Instead, participants were instructed to share either fake or publicly available information by, for example, sharing URLs of news websites, posting images of celebrities, or liking and following verified Instagram accounts. As the final data corpus does not contain any personal information it is publicly available at http://doi.org/10.5281/zenodo.4472606.

PII	File	N	Count	Proportion
Textual				
Username	comments.json	10	261	0.03
	connections.json	10	1222	0.14
	likes.json	10	883	0.10
	media.json	10	43	0.00
	messages.json	10	2947	0.33
	profile.json	10	10	0.00
	saved.json	11	6	0.00
	searches.json	11	314	0.04
	seen_content.json	11	3144	0.35
	shopping.json	11	1	0.00
	stories_activities.json	11	35	0.00
	Total	**115**	**8866**	**1.00**
Name	comments.json	10	105	0.18
	media.json	10	54	0.09
	messages.json	10	427	0.72
	profile.json	10	10	0.02
	Total	**40**	**596**	**1.00**
Email	comments.json	10	28	0.13
	media.json	10	28	0.13
	messages.json	10	152	0.70
	profile.json	10	10	0.05
	Total	**40**	**218**	**1.00**
Phone	comments.json	10	29	0.16
	media.json	10	9	0.05
	messages.json	10	140	0.79
	Total	**30**	**178**	**1.00**
URL	comments.json	10	1	0.00
	messages.json	10	267	0.96
	profile.json	10	10	0.04
	Total	**30**	**278**	**1.00**
Visual				
PII	**Folder**	**.JPG**	**.MP4**	**Proportion**
Username	photos	49	-	0.11
	stories	255	105	0.84
	videos	-	21	0.05
	Total	**304**	**126**	**1.00**
Face	direct	20	-	0.01
	photos	1046	-	0.67
	stories	290	163	0.29
	videos	-	36	0.02
	Total	**1356**	**199**	**1.00**

Table 3

Descriptive statistics of visual and textual content in the generated Instagram DDP validation corpus



{'media_comments': [['2020-10-28T12:33:03+00:00', 'Call me at 063458769 Phone ', 'egelliefhebber Username '], ['2020-10-23T10:46:07+00:00', '@snowecho212 Username https://www.theguardian.com/science/2020/oct/12/researchers-crack-question-of-whether-couples-start-looking-alike. Replicability crisis? 😢', 'egelliefhebber Username '], ['2020-10-21T09:24:23+00:00', "Bit scared, I called Theo Name but he didn't pick up. Maybe you could tell him to call me back at 0623234343 Phone ", 'insta4dummy Username '], ['2020-10-19T17:53:36+00:00', 'Cool! Can you call me at +31612345678 Phone ?', 'horsesarecool52 Username '], ['2020-10-16T10:46:36+00:00', 'Ik kan ook! Nodig mij ook uit op franshansen@gmail.com Email ', 'kippie_toktok Username '], ['2020-10-16T10:44:15+00:00', 'Nice mask, where can I buy these nice fitting masks?', '100billionfaces Username '], ['2020-10-15T11:52:45+00:00', 'Beautiful chicken 😍', 'kippie_toktok Username ']]}

Fig. 1. An example of how labeling a comments.json file would look like in Label-Studio.

3.3. Annotating validation corpus

3.3.1. Textual content

A human rater manually annotated the text files of the validation corpus, labelling all PII occurrences per DDP[2]. The PII were categorized into usernames, first names, phone numbers, e-mail addresses, and URLs that linked to a personal Instagram account. To make the counting of the labels more efficient and less prone to errors, the labeling process was done in Label-Studio (Figure 1). Label-Studio returns an output file (result.json) that consists of one dictionary per file (e.g., 'messages.json') per package (e.g., '100billionfaces_20201021'). Each dictionary contains the labeled PII (e.g., 'horsesarecool52') and corresponding labels (e.g., 'Username') for that particular file.

After this *ground truth* was established, the number of PII occurrences per text file, per DDP could be determined. As can be seen in Table 5, the PII frequency varies highly per file. For example, approximately 72% of all first names present in the entire validation corpus were found in messages.json files only.

3.3.2. Visual content

To annotate visual content, a procedure was carried out by hand. For each media file, it was determined whether there were one or multiple identifiable faces present. To determine whether a face was identifiable, we used a pragmatic definition where we defined a face as identifiable if at least three out of five facial landmarks were visible (right eye, left eye, nose, right mouth corner and left mouth corner) [38].

[2] N.B. Establishing this *ground truth* only has to be done once. The labeling output, together with the 11 Instagram DDPs, are publicly available.

4. Method

~~In this section we describe the approach and implementation of our de-identification algorithm. The developmental corpus for our algorithm is a small set of DDPs downloaded by the researchers. Although this data-set was small, we could already see a lot of variety in structure and content providing a useful basis for developing and testing our de-identification approach. All software is written in python and publicly available at https://github.com/UtrechtUniversity/anonymize-ddp.~~

4.1. ~~Approach~~

~~To de-identify a number of Instagram DDPs, three main steps are undertaken per DDP (see also Figure 2):~~To de-identify a set of collected Instagram DDPs, the algorithm performs three steps on each DDP of the collected set separately (Figure 2):

(1) Pre-process DDP
(2) De-identify text files:

- Detecting PII in (structured) text
- ~~Replacing~~De-identify PII with corresponding de-identification codes

(3) De-identify media files by detecting and blurring human faces and text

4.2. Pre-processing

The software consists of a wrapper and de-identification algorithms. The wrapper handles the pre-processing of the DDP and contains steps specific for Instagram. It unpacks the DDP and removes all files that are not considered relevant for social science research, like “autofill.json" and “account history.json". The user’s profile “profile.json" is de-identified separately in this pre-processing phase, as its content and structure deviate from the other text files in the DDP. After the DDP is cleaned, the PII should be extracted.

4.3. De-identify text files

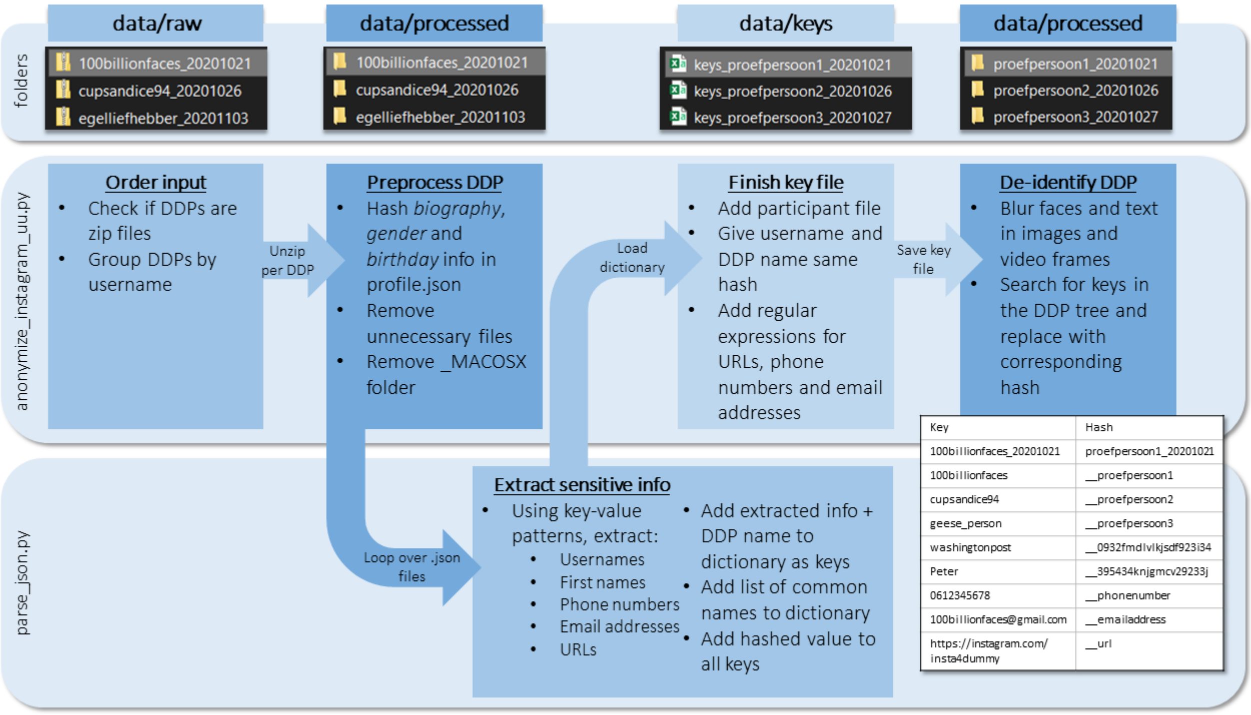
4.3.1. Detecting PII in (structured) text

~~All text files in an Instagram DDP contain a nested structure of keys and values (see Figure 3). To extract PII from these texts, we have determined which key and value combinations and patterns are indicative for PII.~~

~~Per .json file, the algorithm is recursively parsed over the nested structure, each time checking if the specific structure matches (1) a label: username value combination, (2) a username label: timestamp value combination, or (3) a list of length X with at least one timestamp and username value.~~

~~To illustrate the first pattern, each conversation between two or more users stored in the “messages.json" file is a dictionary, containing multiple sub-dictionaries per sent message. Within this ‘smallest structure’ there is always a label ‘sender’ followed by the username. The algorithm will look for ‘sender’ and other similar standard labels. When the corresponding value matches a username (i.e., a string between 3 and 30 elements without special characters except underscores or points), it will be added to the dictionary.~~

~~The second situation can be found in the “connections.json" file, a dictionary with multiple types of connection labels (e.g., ‘close_friends’). Subsequently, each label is made up of another dictionary with~~


folders
data/raw
100billionfaces_20201021
cupsandice94_20201026
egelliefhebber_20201103
data/processed
100billionfaces_20201021
cupsandice94_20201026
egelliefhebber_20201103
data/keys
keys_proefpersoon1_20201021
keys_proefpersoon2_20201026
keys_proefpersoon3_20201027
data/processed
proefpersoon1_20201021
proefpersoon2_20201026
proefpersoon3_20201027
anonymize_instagram_uu.py
Order input
Check if DDPs are zip files
Group DDPs by username
Unzip per DDP
Preprocess DDP
Hash *biography*, *gender* and *birthday* info in profile.json
Remove unnecessary files
Remove _MACOSX folder
Load dictionary
Finish key file
Add participant file
Give username and DDP name same hash
Add regular expressions for URLs, phone numbers and email addresses
Save key file
De-identify DDP
Blur faces and text in images and video frames
Search for keys in the DDP tree and replace with corresponding hash
parse_json.py
Loop over .json files
Extract sensitive info
Using key-value patterns, extract:
Usernames
First names
Phone numbers
Email addresses
URLs
Add extracted info + DDP name to dictionary as keys
Add list of common names to dictionary
Add hashed value to all keys

Key	Hash
100billionfaces_20201021	proefpersoon1_20201021
100billionfaces	__proefpersoon1
cupsandice94	__proefpersoon2
geese_person	__proefpersoon3
washingtonpost	__0932fmdlvlkjsdf923i34
Peter	__395434knjgmcv29233j
0612345678	__phonenumber
100billionfaces@gmail.com	__emailaddress
https://instagram.com/ insta4dummy	__url

Fig. 2. The ~~software~~algorithm takes a zipped DDP as input. Looping over the text (.json) files, all unique instances of PII are detected in the structured part of the data using pattern- and label recognition. The extracted info, together with the most common Dutch first names and, optionally, the participant file, is added to a key file. All occurrences of the keys in the DDP will be replaced with the corresponding hash. Finally, occurrences of human faces and text in media files are detected and blurred. The ~~software~~algorithm will return a de-identified copy of the DDP in the output folder.

~~all corresponding usernames as labels and timestamps (moment of connection) as values. If the label matches a username and the value a timestamp, the username labels will be saved to the dictionary.~~

~~Finally, an example of the (most occurring) third option is the "comments.json" file. Here you have the various commenting labels (e.g., 'media_comments'), each containing a list of lists. The smallest structure in this file is a list with the time of the comment, the comment, and the username of the owner of the media. After checking if one of the items is a timestamp, the algorithm checks if one of the other items matches a username pattern. If this is the case, the username will be added to the dictionary.~~

~~It should be noted that there is also a fourth way of extracting usernames. Even though most usernames found in Instagram DDPs match the above described patterns, usernames can also be mentioned in free text. In this case, there is no standard pattern to look for. Therefore, using regular expressions, the algorithm will search for tagged people (i.e., '@username') and shared media (i.e., 'Shared username's story') using regular expressions.~~

~~Similar to usernames, the text files are checked for patterns (i.e., 'label: PII') and free text indicative of email-addresses and phone numbers. Different from extracting usernames, the regular expressions used to find email-addresses, phone numbers, and URLs are not applied in the 'PII-identifying phase', but are explicitly added to the final dictionary. This way, not all occurrences will be added to the dictionary, increasing its size and reducing the efficiency (during the de-identification phase (see below), the algorithm needs to look for each key separately). Instead, by only adding the search patterns to the dictionary, the de-identification process remains efficient and becomes more inclusive. An important~~

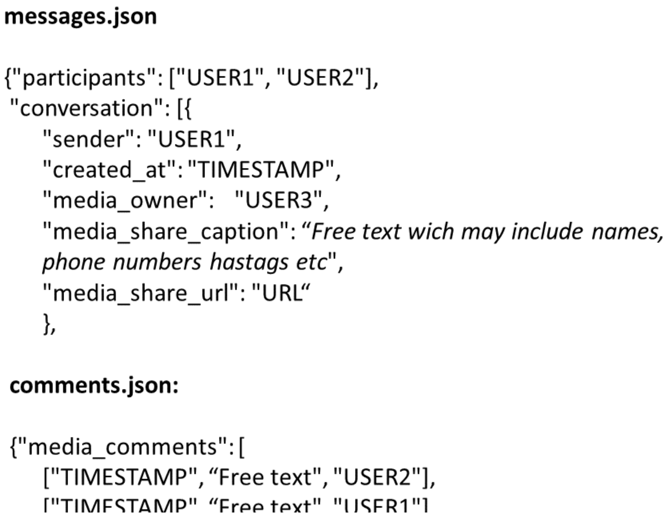
		Structured		**Unstructured**	
Category	**Description**	**Detection method**	**Example**	**Detection method**	**Example**
Name	First names	-	-	List of 1000 most common Dutch names (is interchangable)	{"text": "Hi Tom, hoe gaat het met jou?"}
Username	Unique name created by the Instagram DDPs owner. Has a minimum length of 3 characters and a maximum of 30. Can exist of letters, numbers, points, and underscores.	key-value pairing: e.g., 'author', 'sender', 'participants'.	{"timestamp": "2020-10-23T11: 16:45+00:00", "author": "kippie_toktok"}	Pattern search: i.e., username tags (i.e., @<username>) or shared stories (i.e., Shared <username>'s story)	{"text": "Hebben jullie @kippie_toktok nog gezien?"} or {"story_share": "Shared kippie_toktok's story"}
Emailadress	emailadress, can contain letters, numbers, letters, numbers, or other 7bit ASCII special characters	key-value pairing: i.e., '*mail'.	{email: "blabla@ kippietok.nl"}	Pattern search: i.e., 'letters/numbers/ special characters @letters/numbers.letters'	{"text": "You can mail me at anne7809 @iclouq.nl"}
Phone number	A phone number can contain numbers, spaces and/or dashes	-	-	Pattern search: i.e., minimum of 6 and maximum of 13 numbers	{"text": "This is my number: 06 123 456 78"}
URL	Only URLs referring to other instagram accounts will be pseudonymized.	-	-	Pattern search: i.e., a string starting with 'https://' followed by letters/numbers/special characters and 'instagram'	{"media_share _url": "https://scontent -atl3-2. cdninstagram. com/v"}

Table 4
Overview of the Personal Identifiable Information (PII) categories and their extraction methods.

~~side note is that the regular expressions will only look for Instagram URLs. This because most of the URLs in the DDPs represent links to public websites. These cannot be traced to an individual person and they might be valuable for social science research. Therefore, these URLs can be left unchanged.~~

~~As (first) names exclusively occur in free text and not in a structured format, it was not possible to systematically extract this type of PII. Therefore, instead of working bottom-up, we applied a top-down approach. After all text files have been checked and the key dictionary is filled, a list of the 10, 000 most common Dutch names is added to this dictionary (which we obtained from the DEDUCE software [4]). Of course, it is also possible to add another list (of another country), making the algorithm applicable in multiple languages.~~

All text files in an Instagram DDP contain a nested structure of keys and values (see Figure 3). To extract PII from these structured parts, we have determined which key-value combinations and which patterns are indicative for each PII category (see Table 4). The algorithm parses over the nested structure



messages.json

```
{"participants": ["USER1", "USER2"],
 "conversation": [{
    "sender": "USER1",
    "created_at": "TIMESTAMP",
    "media_owner":   "USER3",
    "media_share_caption": “Free text wich may include names,
    phone numbers hastags etc",
    "media_share_url": "URL“
    },
```

comments.json:

```
{"media_comments": [
    ["TIMESTAMP", “Free text", "USER2"],
    ["TIMESTAMP", “Free text", "USER1"]
```

Fig. 3. Example of key-value structure in .json files with structured and unstructured text.

in each text file in the DDP. Here, it searches the key-value combinations and patterns. By doing this, it extracts the PII. All detected PII instances are added to the key file.

Part of the PII instances in the DDP are not found in the structured part but do appear in the free text. These PII instances include names, phone numbers, and URLs, but also usernames, for example tagged people ‘@username’. We use regular expressions to detect these PII instances. The free text is parsed to detect individual usernames which are then added to the key file. For email-addresses, phone numbers, and URLs we directly add the regular expression to the key file, as this will increase the performance of the de-identification algorithm.

An important side note is that the regular expressions will only look for Instagram URLs that link to users’ personal pages. The remaining URLs in the DDP are left unchanged, as these represent links to public websites, which cannot be traced to individual users and which may be valuable for social science research.

As (first) names exclusively occur in free text and not in a structured format, it was not possible to systematically extract this type of PII. Therefore, instead of working bottom-up, we apply a top-down approach. After all text files are parsed and the key dictionary is filled, a list of the $10,000$ most common Dutch names is added to this dictionary (which we obtained from the DEDUCE software [4]). Of course, it is also possible to add another list (of another country), making the algorithm applicable in multiple languages.

4.3.2. *De-identifying PII in text files*

~~After the PII is extracted and added to the dictionary, a PII specific de-identification needs to be added. Usernames and names receive a unique hexadecimal code. Note that the same name will always receive the same code. This way it is still possible to perform a network analysis after anonymization is complete. Additionally, it is also possible to provide the algorithm with a list of (user)names (and/or other information) and specific their corresponding codes yourself. This might be interesting for scientific research in which the (user)names of participants need to be (clearly) distinguishable from other~~

~~(user)names. In short, (user)names are pseudonymized as they all receive their own specific code and can, therefore, be reverted back if the dictionary is saved. It is up to the user to decide if this dictionary is saved. On the other hand, email-addresses, phone numbers and URLs will anonymized, as they will be hashed with the general '__emailaddress', '__phonenumber', and '__url' codes, respectively.~~

~~For each DDP, the algorithm will look per PII listed in the dictionary for its occurrences, and replace it with the corresponding de-identification code. The replacement extends from file content to file/folder names, resulting in an entirely de-identified DDP.~~

After all PII are extracted, PII specific pseudonyms are added to the key file. Usernames and names receive a unique hexadecimal code, while email-addresses, phone numbers and URLs will be hashed with the general '__emailaddress', '__phonenumber', and '__url' codes, respectively. Note that the same (user)name will always receive the same code. This way it is still possible to perform a network analysis after de-identification is complete.

Additionally, it is possible to provide the algorithm with a list of (user)names (and/or other information) and your own corresponding pseudonyms. This might be interesting for scientific research in which the (user)names of participants need to be distinguishable from other (user)names.

When the key file is complete, the algorithm will parse over the listed PII, search for any occurrences in the entire DDP and replace them with the corresponding pseudonyms. The replacement is also performed on the file/folder names, resulting in an entirely de-identified DDP. There is also an option to save the key file, making it possible to (partly) decode the DDP.

4.4. *De-identifying PII in media*

Besides being able to link textual data to specific individuals, individuals may also be identified by their presence in the images or videos in a DDP. In addition, the images or videos can contain text which may include usernames, person names or other sensitive information. We detect faces in visual content using multi-task Cascaded Convolutional Networks [38] in Facenet Pytorch [27] and blur all occurrences using the Python Imaging Library [39]. We detect text using a pre-trained [40] EAST text detection model [41] and blur all occurrences using the Gaussian blur option provided by OpenCV [42].

4.5. *Evaluation approach*

The developed de-identification procedure is applied to the annotated validation corpus, using the options of applying participant codes for a selected group of users and capital sensitivity for first names.

5. ~~Evaluation~~

5.1. *soutData-set*

~~To evaluate the performance of the software introduced in the Methodology Section, a group of 11 participants generated Instagram DDPs by actively using a new Instagram account for approximately a week. Here, the participants followed guidelines instructing them to actively generate the type of information that the software aims to de-identify.~~

~~The participants were instructed not to share any of their personal information via the Instagram accounts. Instead, participants were instructed to share either fake or publicly available information, such as URLS of news websites, images of celebrities or likes and follows of verified Instagram~~

~~accounts. As the final data-set does not contain any personal information it is publicly available at http://doi.org/10.5281/zenodo.4472606.~~

Visual							
		Direct	Photos	Profile	Stories	Videos	Total
Files							
	.JPEG	11	525	11	176	-	723
	.MP4	-	-	-	92	15	107
Faces							
	.JPEG	20	1046	-	290	-	1,356
	.MP4	-	-	-	163	36	199
Usernames							
	.JPEG	-	49	-	255	-	304
	.MP4	-	-	-	105	21	126
Textual							
	DDP_id	E-mail	Name	Phone	URL	Username	Total
comments.json	-	28	105	29	1	261	424
connections.json	-	-	-	-	-	1,222	1,222
likes.json	-	-	-	-	-	883	883
media.json	-	28	54	9	-	43	134
messages.json	294	152	421	139	267	2,659	3,932
profile.json	18	10	-	-	10	1	39
saved.json	-	-	-	-	-	6	6
searches.json	-	-	-	-	-	314	314
seen_content.json	-	-	-	-	-	3,143	3,143
shopping.json	-	-	-	-	-	1	1
stories_activities.json	-	-	-	-	-	35	35
total	312	218	580	177	278	8,568	10,133

Table 5

~~Descriptive statistics of visual and textual content in the generated Instagram DDP data-set~~

~~The final data-set comprised 11 Instagram DDPs, containing a total of 723 .JPEG files (images) on which $1,336$ faces were identified and 304 usernames and 107 videos on which 164 faces were identified and 126 usernames. In addition, the .json files contain $8,866$ usernames, 904 first names, 218 e-mail addresses, 178 phone numbers and 278 URLS. See Table 5 for more detailed descriptive statistics regarding the visual content of the generated Instagram DDPs data-set.~~

5.2. ~~*Approach for textual content*~~

~~To evaluate the performance of the de-identification procedure in terms of textual content we consider PII in the form of usernames, first names, e-mail addresses, phone numbers and URLS.~~

~~The first step of the evaluation procedure is establishing a *ground truth*. Using the 11 Instagram DDPs, a human rater had to manually label all PII categories per text file, per DDP[3]. To make the counting of the labels more efficient and less prone to errors, the labeling process was done in Label-Studio (Figure 1).~~

~~Label-Studio returns an output file (result.json) that consists of multiple dictionaries; one per file (e.g., ‘messages.json’), per package (e.g., ‘100billionfaces_20201021’). These dictionaries contain all~~

~~the labeled text-items (e.g., ‘horsesarecool52’) and corresponding labels (e.g., ‘Username’) present in that specific file (Figure 4).~~

~~Based on the ground truth, the number of PII categories per text file, per DDP can be determined. Next, using the key files created in the de-identification process, the number of corresponding hashes present in the de-identified DDPs are also calculated per text file, per DDP.~~

~~Comparing the PII occurences in the raw DDPs with the PII and corresponding hash occurences, the software can determine the number of times a type of PII was correctly de-identified (True Positive, TP), the number of times a piece of text was incorrectly de-identified (False Positive, FP) and the number of times PII was not de-identified (False Negative, FN). Finally, the recall-, precision-, and F1-score are calculated.~~

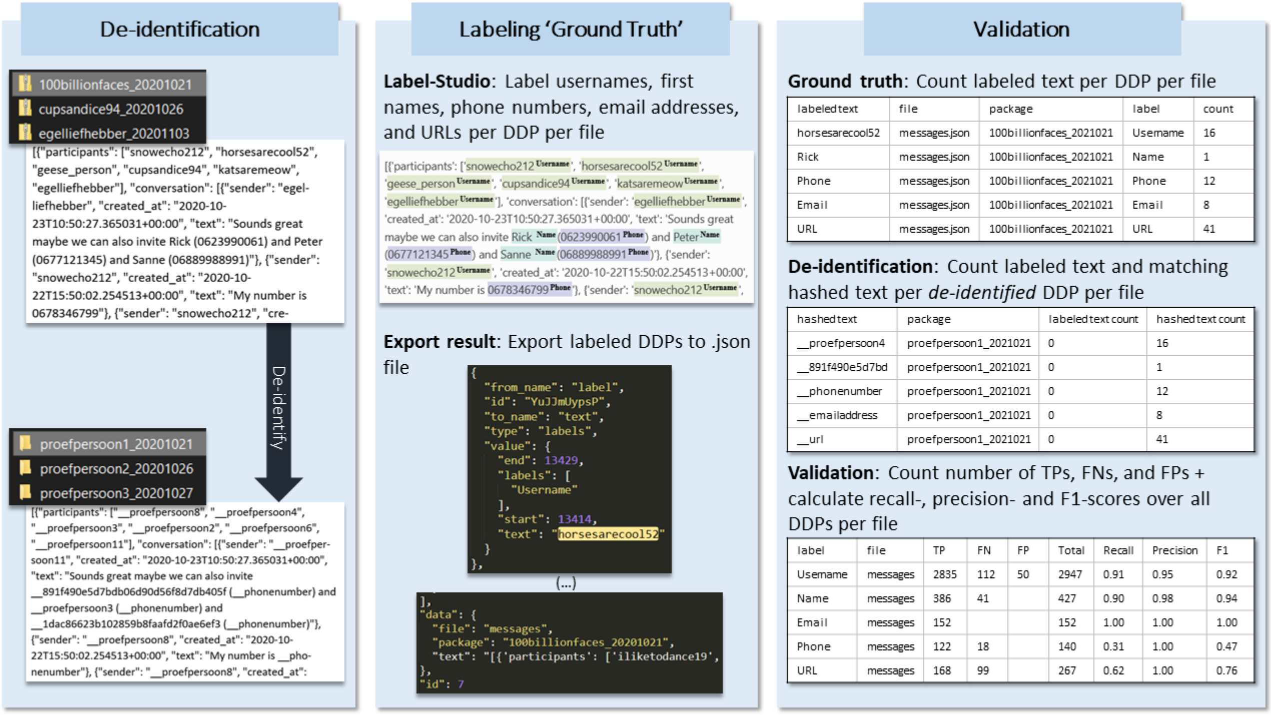
~~The username is the most important type of PII in DDPs, this holds for Instagram but for DDPs of many other data controllers as well, as usernames are typically unique and can be related to the data subject directly. The software distinguishes between two types of usernames. The researcher can provide a list with usernames of all research participants, and these usernames should be replaced with participant numbers (first type). The second type are all other usernames that appear in the DDPs and those should be replaced by a unique identification code. For both types it holds that they can by correctly de-identified (TP), not be de-identified (FN) or a random piece of text can be replaced by the participant number of the hash (FP). In addition, when a username of a participant is replaced by a wrong participant number or a unique identification code, this is also considered a FN. Researchers intended to use this software can decide for themselves if they want to include a list with participants.~~

~~First names should be replaced by a unique identification code (TP). If first names are not replaced they are flagged as falve negatives. In addition, false positives can occur, for example if a hash is applied to a word that is mistaken for a first name, such as the word “ben" in the Dutch sentence “Ik ben vandaag jarig." In addition to the list containing the 10.000 most frequently used Dutch first names that has been used in the EHR de-identification software DEDUCE [4], we added the first names of the research participants to the list. Furthermore, the software allows you to decide if you want to hash only names that appear in the names list and that start with a capital in the DDP, or if you also want to hash names that do not start with a capital.~~

5.3. ~~*Approach for visual content*~~

~~To annotate visual content, a procedure was carried our by hand, as for each file it had to be determined whether there were one or multiple identifiable faces present and for each detected face whether it was indeed de-identified by the software. To determine whether a face was identifiable, we used a pragmatic definition where we defined a faces as identifiable if at least three out of five facial landmarks were visible (right eye, left eye, nose, right mouth corner and left mouth corner) [38]. This definition will not hold if a person will for example actively try to identify individuals by combining multiple images where a person is partly visible, but it provides a sufficient quality in the sense that in case of a data leak, the person on the images is not directly identified.~~

~~For each piece of visual content it holds that each identified face is considered a single observation which can be either appropriately de-identified (TP) or not (FN). Note that although a video consists of multiple frames in which the possibility arises that a face is identifiable, an instance of one frame showing an identifiable face following our definition results in one FN for this face in the movie. As the determination of whether a face is defined identifiable or not is performed by a human rater and this distinction is sometimes not straightforward, the questionable cases are independently rated by two~~



100billionfaces_20201021
cupsandice94_20201026
egelliefhebber_20201103

```
[{"participants": ["snowecho212", "horsesarecool52",
"geese_person", "cupsandice94", "katsaremeow",
"egelliefhebber"], "conversation": [{"sender": "egel-
liefhebber", "created_at": "2020-10-
23T10:50:27.365031+00:00", "text": "Sounds great
maybe we can also invite Rick (0623990061) and Peter
(0677121345) and Sanne (06889988991)"}, {"sender":
"snowecho212", "created_at": "2020-10-
22T15:50:02.254513+00:00", "text": "My number is
0678346799"}, {"sender": "snowecho212", "cre-
```

proefpersoon1_20201021
proefpersoon2_20201026
proefpersoon3_20201027

```
[{"participants": ["__proefpersoon8", "__proefpersoon4",
"__proefpersoon3", "__proefpersoon2", "__proefpersoon6",
"__proefpersoon11"], "conversation": [{"sender": "__proefper-
soon11", "created_at": "2020-10-23T10:50:27.365031+00:00",
"text": "Sounds great maybe we can also invite
__891f490e5d7bdb06d90d56f8d7db405f (__phonenumber) and
__proefpersoon3 (__phonenumber) and
__1dac86623b102859b8faafd2f0ae6ef3 (__phonenumber)"},
{"sender": "__proefpersoon8", "created_at": "2020-10-
22T15:50:02.254513+00:00", "text": "My number is __pho-
nenumber"}, {"sender": "__proefpersoon8", "created_at":
```

Label-Studio: Label usernames, first names, phone numbers, email addresses, and URLs per DDP per file

```
[{'participants': ['snowecho212 Username', 'horsesarecool52 Username',
'geese_person Username', 'cupsandice94 Username', 'katsaremeow Username',
'egelliefhebber Username'], 'conversation': [{'sender': 'egelliefhebber Username',
'created_at': '2020-10-23T10:50:27.365031+00:00', 'text': 'Sounds great
maybe we can also invite Rick Name (0623990061 Phone) and Peter Name
(0677121345 Phone) and Sanne Name (06889988991 Phone)'}, {'sender':
'snowecho212 Username', 'created_at': '2020-10-22T15:50:02.254513+00:00',
'text': 'My number is 0678346799 Phone'}, {'sender': 'snowecho212 Username',
```

Export result: Export labeled DDPs to .json file

```
{
  "from_name": "label",
  "id": "YuJJmUypsP",
  "to_name": "text",
  "type": "labels",
  "value": {
    "end": 13429,
    "labels": [
      "Username"
    ],
    "start": 13414,
    "text": "horsesarecool52"
  }
},
```

```
],
"data": {
  "file": "messages",
  "package": "100billionfaces_20201021",
  "text": "[{'participants': ['iliketodance19',
},
"id": 7
```

Ground truth: Count labeled text per DDP per file

labeledtext	file	package	label	count
horsesarecool52	messages.json	100billionfaces_2021021	Usemame	16
Rick	messages.json	100billionfaces_2021021	Name	1
Phone	messages.json	100billionfaces_2021021	Phone	12
Email	messages.json	100billionfaces_2021021	Email	8
URL	messages.json	100billionfaces_2021021	URL	41

De-identification: Count labeled text and matching hashed text per *de-identified* DDP per file

hashedtext	package	labeledtext count	hashedtext count
__proefpersoon4	proefpersoon1_2021021	0	16
__891f490e5d7bd	proefpersoon1_2021021	0	1
__phonenumber	proefpersoon1_2021021	0	12
__emailaddress	proefpersoon1_2021021	0	8
__url	proefpersoon1_2021021	0	41

Validation: Count number of TPs, FNs, and FPs + calculate recall-, precision- and F1-scores over all DDPs per file

label	file	TP	FN	FP	Total	Recall	Precision	F1
Usemame	messages	2835	112	50	2947	0.91	0.95	0.92
Name	messages	386	41		427	0.90	0.98	0.94
Email	messages	152			152	1.00	1.00	1.00
Phone	messages	122	18		140	0.31	1.00	0.47
URL	messages	168	99		267	0.62	1.00	0.76

Fig. 4. The raw DDPs in which all PII categories are labeled (i.e., the ground truth) is compared with the de-identified DDPs. The ~~software~~algorithm counts the number of PII categories (total), correctly hashed PII (TP), falsely hashed information (FP), and unhashed PII (FN). Subsequently, a recall-, precision-, and F1-score can be calculated.

~~raters and classification is performed based on consensus. In addition, a set of 100 .JPEG files and 20 .MP4 files were independently annotated by two separate annotators. On the .JPEG files, 204 faces were identified and from these, 193 were identified by both raters, which equals 94.6%. On this subset, a Cohen's κ inter-rater reliability was calculated of 1, so the raters highly agreed on which faces were appropriately de-identified and which not. For the .MP4 files, 49 faces were identified and from these, 41 were identified by both raters, which equals 83.7%. On this subset, a Cohen's κ inter-rater reliability was calculated of 0.62. The sample of faces was much smaller for .MP4 compared to .JPEG, and it was apparently also a lot more difficult to determine whether a face was appropriately identified when the image was moving compared to when it was a still image.~~

~~In addition, particularly on Instagram, visual content can contain usernames. The software is not able to distinguish between usernames and other types of text, and therefore usernames on visual content can only be detected and de-identified, distinctions between research participants and other usernames are not made. Therefore, appropriately de-identified usernames are counted as true positives (TP) and usernames not de-identified are counted as false negatives (FN). False positives cannot be quantified in the current procedure.~~

5.3.1. *Textual content*

The effectiveness of the de-identification performance on textual content is assessed by determining the number of times PII has been correctly de-identified (True Positive, TP), incorrectly de-identified (False Positive, FP), and not de-identified (False Negative, FN)(4). Using these statistics, the recall-, precision-, and F1-score are calculated.

5.3.2. Visual content

The human rater determined for each detected face whether it was indeed de-identified by the ~~software~~algorithm. The definition of identifiable used (i.e., at least three out of five facial landmarks were visible [38]), will not hold if, for example, a person will actively try to identify individuals by combining multiple images where a person is partly visible. However, it is sufficient for the level of de-identification we are currently aiming at.

For each piece of visual content an identified face is considered a single observation which can be either appropriately de-identified (TP) or not (FN). Note that although a video consists of multiple frames in which the possibility arises that a face is identifiable, an instance of one frame showing an identifiable face following our definition results in one FN for this face in the movie.

As the determination of whether a face is defined identifiable or not is performed by a human rater and this distinction is sometimes not straightforward, the questionable cases are independently rated by two raters and classification is performed based on consensus. In addition, a set of 100 .JPEG files and 20 .MP4 files were independently annotated by two separate annotators.

On the .JPEG files, 204 faces were identified and from these, 193 were identified by both raters. On this subset, a Cohen's κ inter-rater reliability was calculated of 1, so the raters highly agreed on which faces were appropriately de-identified and which were not. For the .MP4 files, 49 faces were identified and from these, 41 were identified by both raters. On this subset, a Cohen's κ inter-rater reliability was calculated of 0.62. The sample of faces was much smaller for .MP4 compared to .JPEG, and it was apparently also a lot more difficult to determine whether a face was appropriately identified when the image was moving compared to when it was a still image.

In addition, particularly on Instagram, visual content can contain usernames. The algorithm is not able to distinguish between usernames and other types of text. therefore all text is de-identified, without distinctions between text and usernames, or without replacing usernames for their key value. Therefore, de-identified usernames are counted as true positives (TP) and usernames not de-identified are counted as false negatives (FN). False positives cannot be quantified in the current procedure.

5.4. Evaluation criteria

~~For each category of PII in each filetype in the set of DDPs regarding textual content, we count the number of TP, FP and FN. For the visual content, we calculate the TP and FN.~~ We use scikit learn to further evaluate the performance of the procedure on the different aspects [43]. First, we calculate the recall, or the sensitivity, as

$$Recall = \frac{TP}{TP + FN}. \qquad (1)$$

Here, we measure the ratio of the correctly de-identified cases to all the cases that were supposed to be de-identified (i.e. ground truth). Each false negative potentially results in not preserving the privacy of a research participant and therefore a high value for the recall is particularly important. The precision is calculated as

$$Precision = \frac{TP}{TP + FP}. \qquad (2)$$

Precision shows the ratio of correctly de-identified observations to the total of de-identified observations and a high precision illustrates that the amount of additional information lost due to unnecessary de-identification is limited. Given that DDPs are typically collected to analyze aspects such as the free text or the images, losing a lot of this information by the de-identification process challenges the intended research goal. At last, we calculate the F1 score

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall}, \tag{3}$$

which combined the precision and recall and considered both false positives and false negatives. Note that we do not calculate the accuracy as the number of true negatives cannot be determined appropriately in our data-set.

6. Results

6.1. Initial Results

~~In Table 6, the results of the application of the software to our Instagram DDP data-set can be found, where we chose for settings including a participant file and capital sensitivity for first names. Regarding the visual content, we can conclude a large proportion of faces on images is appropriately detected and blurred, while on videos this proportion is substantively lower. Apparently, faces are harder to detect by the detection algorithm when the images are moving.~~

~~Regarding textual content, we can conclude that email addresses are appropriately detected and anonymized throughout all files within the DDPs. Regarding names, phone numbers and URLs, we can conclude that a substantial amount of names are not detected by the algorithm throughout the different files. The quality of the anonymization of usernames differs a lot depending on the file. Only in the file 'messages.json', false positives are detected. Furthermore, relatively lower recall values are measured for the files 'media.json' and 'saved.json', although these files have a small number of total observations.~~

~~By critically investigating the results found in Table 6, and investigating what coding decisions led to the most (negatively) outstanding results, improvements to the code were made.~~

A large proportion of faces on images were appropriately detected and blurred (Table 6), while on videos this proportion was substantively lower. Apparently, faces are harder to detect by the algorithm when the images are moving.

Email addresses were appropriately detected and de-identified throughout all files within the DDPs (Table 6), whereas a substantial amount of names were not detected by the algorithm throughout the different files. The quality of the de-identification of usernames differs a lot depending on the file. False positives were only detected in the 'messages.json' file. Furthermore, relatively lower recall values were measured for the files 'media.json' and 'saved.json', although these files have a small number of total observations.

The annotated validation corpus contains both Dutch and English text; some within the same document. We observed no difference between de-identification of PII in English and Dutch text.

By critically examining the results of Table 6 and investigating what coding decisions led to the least optimal results, improvements to the code were made.

Visual								
		Total	TP	FN	FP	Recall	Precision	F1
Faces								
	.JPEG	1,356	1,205	151	-	0.89	-	-
	.MP4	199	131	68	-	0.66	-	-
	Total	1,555	1,336	219	-	0.86	-	-
Usernames								
	.JPEG	304	302	2	-	0.99	-	-
	.MP4	126	125	1	-	0.99	-	-
	Total	430	427	3	-	0.99	-	-
Textual								
	file	total	TP	FN	FP	Recall	Precision	F1
Email								
	comments.json	28	28	0	0	1	1	1
	media.json	28	28	0	0	1	1	1
	messages.json	152	152	0	0	1	1	1
	profile.json	10	10	0	0	1	1	1
	total	218	218	0	0	1	1	1
Name								
	comments.json	105	61	44	0	0.5619	0.9365	0.7024
	media.json	54	41	13	0	0.7593	1	0.8530
	messages.json	427	386	41	0	0.9040	0.9836	0.9374
	profile.json	10	6	4	0	0.6	1	0.75
	total	596	494	102	0	0.8255	0.9798	0.8936
Phone								
	comments.json	29	26	3	0	0.4828	1	0.6512
	media.json	9	7	2	0	0.4444	1	0.6154
	messages.json	139	121	18	0	0.3022	1	0.4641
	total	177	154	23	0	0.3390	1	0.5063
URL								
	comments.json	1	0	1	0	0	0	0
	messages.json	267	168	99	0	0.6180	1	0.7639
	profile.json	10	10	0	0	1	1	1
	total	278	178	100	0	0.6295	1	0.7726
Username								
	comments.json	261	252	9	0	0.9655	1	0.9813
	connections.json	1,222	1,190	32	0	0.9722	1	0.9858
	likes.json	883	823	60	0	0.9320	1	0.9611
	media.json	43	33	10	0	0.7674	0.7907	0.7788
	messages.json	2,947	2,835	112	50	0.9067	0.9500	0.9196
	profile.json	10	10	0	0	1	1	1
	saved.json	6	4	2	0	0.6667	1	0.8
	searches.json	314	305	9	0	0.9713	1	0.9855
	seen_content.json	3,144	2,619	525	0	0.8330	0.9876	0.8931
	shopping.json	1	1	0	0	1	1	1
	stories_activities.json	35	34	1	0	0.9714	1	0.9851
	total	8,866	8,106	760	50	0.89567	0.9775	0.9324

Table 6

Results in terms of TP, FP, FN, recall, precision and F1.

6.2. ~~Further improvements~~Code adjustments

~~The first improvement relates to the 'profile.json' file. Here, the entire entry that can be found after 'name' is now added to the key file and the similar key is used for the DDP username. In this way, the participant can be recognized throughout the complete DDP with either their username of their name. A second improvement was made after further inspecting the relatively large amount of false positives in the 'seen_content.json' file. Based on this, the list of labels that should be exempted from hashing has been extended. Based on a more thorough inspection of the type of usernames that were not detected by the algorithm, the username format has been adjusted in such a way that usernames are detected as such when they contain at least three characters, the minimum limit in the previous version of the code as six characters. After further inspecting the false positive first names, the names 'Van', 'Door' and 'Can' were removed from the list with the $10,000$ most frequently used first names because they also represent words commonly used in free text, resulting in a lot of FPs. At last, the hash function for usernames became case insensitive, as Instagram does not distinguish between lowercases and uppercases in usernames, while the software initially generated a different hash as an uppercase was used somewhere in the username compared to the username without uppercase.~~

~~The improved script has fewer false negatives regarding names, phone numbers and URLS. Regarding usernames, both the number of false negatives and false positives has decreased substantively.~~

First of all, we made some changes to how the the 'profile.json' file was processed. This change implied adding the entire entry that can be found after the key 'name' to the key file, receiving the same pseudonym as used for the DDP username. This way, participants can now be recognized throughout the de-identified DDP by both their masked username *and* their (first) name. After the adjustment, these 'profile' names and DDP usernames are labeled as 'DDP_id', resulting in a shift in the initial username and name frequencies (see Table 7).

A second improvement has been made after further inspecting the relatively large amount of false positives in the 'seen_content.json' file. Based on this, the list of labels that should be exempted from hashing has been extended.

Third, based on a more thorough inspection of the type of usernames that were not detected by the algorithm, the username format has been adjusted in such a way that usernames are detected as such when they contain at least three characters. The minimum limit in the previous version of the code was six characters.

After further inspecting the false positive first names, the names 'Van', 'Door' and 'Can' were removed from the list with the $10,000$ most frequently used first names because they also represent words commonly used in free text, resulting in a lot of FPs.

At last, the hash function for usernames became case insensitive, as Instagram does not distinguish between lower cases and upper cases in usernames. Initially, the algorithm generated a different hash as an uppercase was used somewhere in the username compared to when the same username was used without an uppercase.

6.3. Final results

The adjusted algorithm was applied to the annotated validation corpus and the de-identification performance on textual was again evaluated. The adjusted algorithm produces fewer false negatives regarding names, phone numbers and URLs (Table 8). Regarding usernames, both the number of false negatives and false positives decreased substantively.

PII	File	N	Count	Proportion
Username	comments.json	10	261	0.03
	connections.json	10	1222	0.14
	likes.json	10	883	0.10
	media.json	10	43	0.01
	messages.json	10	2659	0.31
	profile.json	10	0	0.00
	saved.json	11	6	0.00
	searches.json	11	314	0.04
	seen_content.json	11	3144	0.37
	shopping.json	11	1	0.00
	stories_activities.json	11	35	0.00
	Total	**115**	**8568**	**1.00**
DDP_id	messages.json	10	294	0.94
	profile.json	10	20	0.06
	Total	**20**	**314**	**1.00**
Name	comments.json	10	105	0.18
	media.json	10	54	0.09
	messages.json	10	427	0.72
	profile.json	10	10	0.02
	Total	**40**	**596**	**1.00**
Email	comments.json	10	28	0.13
	media.json	10	28	0.13
	messages.json	10	152	0.70
	profile.json	10	10	0.05
	Total	**40**	**218**	**1.00**
URL	comments.json	10	1	0.00
	messages.json	10	267	0.96
	profile.json	10	10	0.04
	Total	**30**	**278**	**1.00**
Phone	comments.json	10	29	0.16
	media.json	10	9	0.05
	messages.json	10	140	0.79
	Total	**30**	**178**	**1.00**

Table 7

Descriptive statistics of textual content in the generated Instagram DDP data corpus after adjustment of the script

7. ~~Conclusions~~Limitations and future work

~~Data Download Packages (DDPs) contain all data collected by public and private entities during the course of citizens' digital life. Although they form a treasure trove for social scientists, they contain data that can be deeply private. To protect the privacy of research participants while they let their DDPs be used for scientific research, we developed de-identification software that is able to anonymize and pseudonymize data that follow typical DDP structures.~~

~~We evaluated the performance of the de-identification software on a set of Instagram DDPs. From this application we could conclude that the software is particularly well suited to anonymize and/or pseudonymize usernames, e-mail addresses and phone-numbers from structured and unstructured text~~

	file	total	TP	FN	FP	Recall	Precision	F1
DDP_id								
	messages.json	294	294	0	0	1	1	1
	profile.json	18	18	0	0	1	1	1
	total	312	312	0	0	1	1	1
E-mail								
	comments.json	28	28	0	0	1	1	1
	media.json	28	28	0	0	1	1	1
	messages.json	152	152	0	0	1	1	1
	profile.json	10	10	0	0	1	1	1
	total	218	218	0	0	1	1	1
Name								
	comments.json	105	98	7	0	0.9333	1	0.9654
	media.json	54	45	9	0	0.8333	1	0.9042
	messages.json	421	385	36	0	0.9145	1	0.9509
	total	580	528	52	0	0.9103	1	0.9519
Phone								
	comments.json	29	29	0	0	1	1	1
	media.json	9	9	0	0	1	1	1
	messages.json	139	138	1	24	0.9928	0.8519	0.9169
	total	177	176	1	24	0.9943	0.88	0.9337
URL								
	comments.json	1	1	0	0	1	1	1
	messages.json	267	267	0	0	1	1	1
	profile.json	10	10	0	0	1	1	1
	total	278	278	0	0	1	1	1
Username								
	comments.json	261	258	3	0	0.9885	1	0.9940
	connections.json	1,222	1,219	3	0	0.9975	1	0.9988
	likes.json	883	881	2	0	0.9977	1	0.9989
	media.json	43	42	1	0	0.9767	1	0.9881
	messages.json	2,659	2,658	1	2	0.9846	0.9868	0.9847
	profile.json	1	1	0	1	0	0	0
	saved.json	6	6	0	0	1	1	1
	searches.json	314	313	1	0	0.9968	1	0.9984
	seen_content.json	3,143	3,137	6	0	0.9981	1	0.9990
	shopping.json	1	1	0	0	1	1	1
	stories_activities.json	35	35	0	0	1	1	1
	total	8,568	8,551	17	3	0.9932	0.9985	0.9952

Table 8

Results in terms of TP, FP, FN, recall, precision and F1 after improvements to the script have been made.

~~files. In addition, it was able to appropriately anonymize faces on .jpg files. Appropriate anonymization and/or pseudonymization of first names appeared more challenging, particularly because some first names can also appear as words in open text and vice versa. However, when applying the software researchers can decide if their focus is on precision or on recall and take measures to accomodate this.~~

~~Furthermore, anonymizing faces on .mp4 files appeared more challenging, typically because in .mp4 files sometimes different parts of faces can be visible at different moments, together providing sufficient information to identify a face, and because Instagram provides in so-called 'filters', which also make it more difficult for the software to detect a face for de-identification.~~

~~The aim of the software was to remove identifiers from DDPs in such a way that research participants cannot be identified when the data is manually investigated or in the undesired situation that someone gains unauthorized access to the data. Appropriate safety measures to prevent this remain required, but based on the results from the validation we do believe that the intended goal of this software is met.~~

~~If researchers intend to use this software for their own research projects, a number of issues should be taken into account. A first issue is that the current script has been primarily be developed to de-identify the Instagram DDPs. However, the software has been written in such a way that with small adjustments it could be applied to DDPs from other data controllers. In future work, we could provide some of these adjustments for specific other data controllers to illustrate how this works in practice, but we also encourage other researchers and software developers to develop such adjustments and share this with the community. A second issue is that, besides adjustments to DDPs from different data controllers, we can also imagine that different researchers might have different research intentions with the collected data and that based on this adjustments to the software might be desired. For example a sociologist with interest in what types of accounts are followed and liked by the research participant might not want to pseudonymize all usernames present in the DDP, but instead only the usernames of the participants for example. A third issue to consider is that if a higher level of security is desired, adjustments can also be made in a quite straightforward manner. For example, it can be chosen not to save the key file or to use hashing and blurring algorithms with higher safety standards.~~

~~An important issue to note further is that because of the fact that faces on images are blurred when this software is used, it is no longer possible to for example apply emotion detection algorithms to the faces on the images in the DDPs under investigation. If emotion detection of faces is a goal of the researcher, it can be considered to replace the blurring part of the software with a procedure that replaces the face with a deepfake of the face [44]. With such an algorithm, it remains possible to detect the emotions on faces, while protecting the privacy of the participants. However, this will inevitably also introduce some noise.~~

~~Another remark regarding the blurring of visual content is that this part of the software could be further developed to be more refined so that it can distinguish between usernames and regular text and that it only blurs the usernames. In addition, it can be further refined in such a way that text written for example at a 45° or 90° is evaluated in a single sequence as well. currently, angled text is typically evaluated in small separate pieces. A last point of attention is that sound in .mp4 files is currently removed. This might be a good thing as it thereby also removes possibly identifying sounds but it might be disadvantageous for certain purposes. Although the use of digital trace data for scientific purposes, and appropriate de-identification of digital trace data are fields that are still at their infancy, our developed software enormously contributes to privacy preserving analysis of digital trace data collected with DDPs.~~

The evaluation results show that the developed algorithm is well-suited to de-identify usernames, e-mail addresses and phone numbers in both structured and unstructured text files. In addition, the algorithm appropriately de-identifies faces on .jpg files. Appropriate de-identification of first names appears more challenging, particularly because some first names are also used as words in free text and vice versa. However, when applying the algorithm, researchers can decide if their focus is on precision or on recall and take measures to accommodate this. Furthermore, de-identifying faces on .mp4 files was more difficult compared to .jpg files. This reduced performance can be explained by the fact that in moving image different parts of faces can be visible at different moments, which provide sufficient information to identify a face when combined. Another reason can be that Instagram provides so-called ‘filters’, which also make it more difficult for the software to detect a face for de-identification.

In terms of generalizability of the developed algorithm, an important first discussion point is the fact that the algorithm has been developed and tested using Instagram DDPs only. As we illustrate in Table 1, the Instagram DDP contains a set of specific features that can be found in DDPs of several other data controllers. Our de-identification approach is designed for these features and therefore we consider it plausible that it can also be applied to DDPs of other data controllers. In general, we think that with small adjustments to the algorithm, high performance levels can be reached relatively straightforwardly when applying the algorithm to DDPs of other data controllers. Such adjustments to the algorithm can be further investigated in future research.

A second point for discussion in terms of generalizability is the fact that data controllers such as social media platforms constantly update their features and develop new ones. Although our algorithm is able to deal with variance in structure and content of DDPs, we envision that small updates may be required when being used on later versions of Instagram DDPs. Third, the de-identification showed good performance on a data-set that was diverse, but limited in size and therefore it is less representative. The algorithm has also been applied in practice to a set of 104 Instagram DDPs as part of the previously described Project AWeSome [6]. Since our method is designed for recognizing text patterns that are specific to DDPs rather than language, it performed well on both English and Dutch text. We believe our approach can easily be applied to DDPs in other languages, which only requires adding a list of common names and possibly adjusting some labels.

Besides generalizability in terms of applications to other data types, the particular research goal should also be considered. For example, if a researcher is interested in the emotions that can be detected on the faces of images in the DDP. This is currently not possible because faces are blurred. In this situation, the researcher can for example replace the blurring algorithm with an algorithm that replaces a face with a deepfake of that face [44]. Alternatively, if a researcher is interest in the type of accounts that are followed and liked by the research participant, it is not desirable to de-identify all usernames in the DDP. In a third example, if a researcher is interested in the the text that is written on the images and videos posted on Instagram, the currently implemented text detection algorithm should be further refined. At this moment, the algorithm does not distinguish between usernames and other types of information written in text and blurs it all. In a last example, a researcher can also be interested in the sound that accompanies videos. In the current version of the algorithm the sound is completely removed.

A last point of discussion considers the safety standards that are currently adhered. We have clearly stated that the algorithm aims to prepare the DDPs in such a way that they can be processed as any other type of sensitive research data, supplemented with other measures such as using shielded (cloud) environments. If the researchers would like to share the data with others on a more flexible level, for example the currently used blurring algorithm is not sufficient as it can be prone to re-identification [36].

8. Conclusion

Data Download Packages (DDPs) contain all data collected by public and private entities during the course of citizens’ digital life. Although they form a treasure trove for social scientists, they contain data that can be deeply private. The privacy of research participants should be protected while they let their

DDPs be used for scientific research, as is the case for all type of sensitive data collected for research. Therefore, we first of all provided an overview of the structure and content of DDPs, both in general and for Instagram in particular, which can serve as a valuable reference for researchers interested using DDPs for future research. For them, our generated DDPs are publicly available. In addition, we developed the first algorithm that is able to de-identify data with DDP structure. Furthermore, we evaluated the performance of this algorithm, which appeared to be of very high level. At last, we provide the algorithm, the validation corpus and the evaluation code open source. Thanks to the GDPR, researchers have the opportunity to collect DDPs with consent from research participants. Now, we have developed an algorithm that also allows researchers to process this data in such a way that is in line with that same GDPR.

References

[1] G.D.P. Regulation, Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46, *Official Journal of the European Union (OJ)* **59**(1–88) (2016), 294. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=OJ:L:2016:119:FULL&from=EN.

[2] G. King, Ensuring the data-rich future of the social sciences, *science* **331**(6018) (2011), 719–721. doi:10.1126/science.1197872.

[3] L. Boeschoten, J. Ausloos, J. Moeller, T. Araujo and D.L. Oberski, Digital trace data collection through data donation, *arXiv preprint arXiv:2011.09851* (2020).

[4] V. Menger, F. Scheepers, L.M. van Wijk and M. Spruit, DEDUCE: A pattern matching method for automatic de-identification of Dutch medical text, *Telematics and Informatics* **35**(4) (2018), 727–736. https://doi.org/10.1016/j.tele.2017.08.002.

[5] G. Coppersmith, M. Mitchell, C. Harman, M. Dredze and R. Leary, Deidentify Twitter, 2017. https://github.com/qntfy/deidentify_twitter.

[6] I. Beyens, J.L. Pouwels, I.I. van Driel, L. Keijsers and P.M. Valkenburg, The effect of social media on well-being differs from adolescent to adolescent, *Scientific Reports* **10**(1) (2020), 1–11.

[7] A. Hundepool and P.-P. De Wolf, Statistical discosure control, *Method Series* (2012), 1–49. https://www.cbs.nl/en-gb/our-services/methods/statistical-methods/output/output/statistical-disclosure-control.

[8] Aticle 29 Data protection working party, Opinion 05/2014 on Anonymisation Techniques, *European Commission* (2014), 1–37. https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf.

[9] J. Trienes, D. Trieschnigg, C. Seifert and D. Hiemstra, Comparing rule-based, feature-based and deep neural methods for de-identification of Dutch medical records, *CEUR Workshop Proceedings* **2551** (2020), 3–11.

[10] J. Simon, Amazon Comprehend Medical–Natural Language Processing 24 for Healthcare Customers, *Retrieved April* **18** (2018), 2019. https://aws.amazon.com/comprehend/medical/.

[11] L. Liu, O. Perez-Concha, A. Nguyen, V. Bennett and L. Jorm, De-identifying Hospital Discharge Summaries: An End-to-End Framework using Ensemble of De-Identifiers, *arXiv preprint arXiv:2101.00146* (2020).

[12] G. Coppersmith, M. Dredze, C. Harman, K. Hollingshead and M. Mitchell, CLPsych 2015 shared task: Depression and PTSD on Twitter, in: *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, 2015, pp. 31–39.

[13] R. Dorn, A.L. Nobles, M. Rouhizadeh and M. Dredze, Examining the Feasibility of Off-the-Shelf Algorithms for Masking Directly Identifiable Information in Social Media Data **1996** (2020). http://arxiv.org/abs/2011.08324.

[14] F. Prasser, F. Kohlmayer, R. Lautenschläger and K.A. Kuhn, Arx-a comprehensive tool for anonymizing biomedical data, in: *AMIA Annual Symposium Proceedings*, Vol. 2014, American Medical Informatics Association, 2014, p. 984.

[15] M. Templ, A. Kowarik and B. Meindl, Statistical disclosure control for micro-data using the R package sdcMicro, *Journal of Statistical Software* **67**(4) (2015). doi:10.18637/jss.v067.i04.

[16] B. van der Sloot, *The General Data Protection Regulation in Plain Language*, Amsterdam University Press, 2020. ISBN 978-94-6372-651-1.

[17] B. Zhong, Y. Huang and Q. Liu, Mental health toll from the coronavirus: Social media usage reveals Wuhan residents' depression and secondary trauma in the COVID-19 outbreak, *Computers in human behavior* **114** (2021), 106524. https://doi.org/10.1016/j.chb.2020.106524.

[18] A. Jungherr, *Analyzing Political Communication with Digital Trace Data: The Role of Twitter Messages in Social Science Research*, Contributions to Political Science, Springer International Publishing, 2015. ISBN 978-3-319-20318-8. doi:10.1007/978-3-319-20319-5. https://www.springer.com/gp/book/9783319203188.

[19] H. Schoen, D. Gayo-Avello, P. Takis Metaxas, E. Mustafaraj, M. Strohmaier and P. Gloor, The power of prediction with social media, *Internet Research* **23**(5) (2013), 528–543. doi:10.1108/IntR-06-2013-0115.

[20] M. Kosinski, D. Stillwell and T. Graepel, Private traits and attributes are predictable from digital records of human behavior, *Proceedings of the National Academy of Sciences* **110**(15) (2013), 5802–5805. doi:10.1073/pnas.1218772110.

[21] L. Sweeney, k-anonymity: A model for protecting privacy, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **10**(05) (2002), 557–570.

[22] K. El Emam and F.K. Dankar, Protecting privacy using k-anonymity, *Journal of the American Medical Informatics Association* **15**(5) (2008), 627–637. https://doi.org/10.1197/jamia.M2716.

[23] C.A. Kushida, D.A. Nichols, R. Jadrnicek, R. Miller, J.K. Walsh and K. Griffin, Strategies for de-identification and anonymization of electronic health record data for use in multicenter research studies, *Medical care* **50**(Suppl) (2012), S82. doi:10.1097/MLR.0b013e3182585355.

[24] P.M. Heider, J.S. Obeid and S.M. Meystre, A Comparative Analysis of Speed and Accuracy for Three Off-the-Shelf De-Identification Tools, *AMIA Summits on Translational Science Proceedings* **2020** (2020), 241. doi:PMCID: PMC7233098.

[25] C. van Toledo, F. van Dijk and M. Spruit, Dutch Named Entity Recognition and De-identification Methods for the Human Resource Domain, *arXiv preprint arXiv:2106.02287* (2021).

[26] M. Azure, Microsoft Azure cognitive services, 2021. https://azure.microsoft.com/nl-nl/services/cognitive-services/face/.

[27] T. Esler, facenet pytorch, 2019. doi:10.34740/KAGGLE/DSV/845275. https://www.kaggle.com/timesler/facenet-pytorch.

[28] R. Nosowsky and T.J. Giordano, The Health Insurance Portability and Accountability Act of 1996 (HIPAA) privacy rule: implications for clinical research, *Annu. Rev. Med.* **57** (2006), 575–590.

[29] Ö. Uzuner, Y. Luo and P. Szolovits, Evaluating the State-of-the-Art in Automatic De-identification, *Journal of the American Medical Informatics Association* **14**(5) (2007), 550–563. doi:10.1197/jamia.M2444.

[30] OpenAIRE, amnesia. https://amnesia.openaire.eu/index.html.

[31] G. Beigi and H. Liu, *A Survey on Privacy in Social Media*, Vol. 1, 2020, pp. 1–38. ISSN 2691-1922. ISBN 0001417126. doi:10.1145/3343038.

[32] G. Beigi, K. Shu, R. Guo, S. Wang and H. Liu, Privacy preserving text representation learning, in: *Proceedings of the 30th ACM Conference on Hypertext and Social Media*, 2019, pp. 275–276.

[33] L. Backstrom, C. Dwork and J. Kleinberg, Wherefore art thou R3579X? Anonymized social networks, hidden patterns, and structural steganography, in: *Proceedings of the 16th international conference on World Wide Web*, 2007, pp. 181–190.

[34] A. Narayanan and V. Shmatikov, Robust de-anonymization of large sparse datasets, in: *2008 IEEE Symposium on Security and Privacy (sp 2008)*, IEEE, 2008, pp. 111–125.

[35] H. Mao, X. Shuai and A. Kapadia, Loose tweets: an analysis of privacy leaks on twitter, in: *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society*, 2011, pp. 1–12.

[36] R. McPherson, R. Shokri and V. Shmatikov, Defeating image obfuscation with deep learning, *arXiv preprint arXiv:1609.00408* (2016).

[37] S. Ribaric, A. Ariyaeeinia and N. Pavesic, De-identification for privacy protection in multimedia content: A survey, *Signal Processing: Image Communication* **47** (2016), 131–151.

[38] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks, *IEEE Signal Processing Letters* **23**(10) (2016), 1499–1503. doi:10.1109/LSP.2016.2603342.

[39] H. van Kemenade, wiredfool, A. Murray, A. Clark, A. Karpinsky, nulano, C. Gohlke, J. Dufresne, B. Crowell, D. Schmidt, A. Houghton, K. Kopachev, S. Mani, S. Landey, vashek, J. Ware, Jason, D. Caro, S. Kossouho, R. Lahd, S. T., A. Lee, E.W. Brown, O. Tonnhofer, M. Bonfill, P. Rowlands, F. Al-Saidi, M. Górny, M. Korobov and M. Kurczewski, python-pillow/Pillow 8.0.0, Zenodo, 2020. doi:10.5281/zenodo.4088798.

[40] O. Yadong, frozen east text detection, 2018. https://github.com/oyyd/frozen_east_text_detection.pb.

[41] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He and J. Liang, EAST: An Efficient and Accurate Scene Text Detector, in: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2642–2651. doi:10.1109/CVPR.2017.283.

[42] G. Bradski, The OpenCV Library, *Dr. Dobb's Journal of Software Tools* (2000). https://opencv.org/.

[43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., Scikit-learn: Machine learning in Python, *the Journal of machine Learning research* **12** (2011), 2825–2830. https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf?source=post_page---------------------------.

[44] P. Korshunov and S. Marcel, Deepfakes: a new threat to face recognition? assessment and detection, *arXiv preprint arXiv:1812.08685* (2018).

[45] H. Hanke and D. Knees, A phase-field damage model based on evolving microstructure, *Asymptotic Analysis* **101** (2017), 149–180.

[46] E. Lefever, A hybrid approach to domain-independent taxonomy learning, *Applied Ontology* **11**(3) (2016), 255–278.

[47] P.S. Meltzer, A. Kallioniemi and J.M. Trent, Chromosome alterations in human solid tumors, in: *The Genetic Basis of Human Cancer*, B. Vogelstein and K.W. Kinzler, eds, McGraw-Hill, New York, 2002, pp. 93–113.

[48] P.R. Murray, K.S. Rosenthal, G.S. Kobayashi and M.A. Pfaller, *Medical Microbiology*, 4th edn, Mosby, St. Louis, 2002.

[49] E. Wilson, Active vibration analysis of thin-walled beams, PhD thesis, University of Virginia, 1991.

[50] S.L. Garfinkel et al., De-identification of personal information, *National institute of standards and technology* (2015).

[51] A. Dehghan, A. Kovacevic, G. Karystianis, J.A. Keane and G. Nenadic, Combining knowledge- and data-driven methods for de-identification of clinical narratives, *Journal of Biomedical Informatics* **58** (2015), S53–S59. doi:10.1016/j.jbi.2015.06.029.