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 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 algorithm that is able to handle typical characteristics of 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 algorithm and illustrate how the algorithm can be tailored towards specific DDP structures.

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

1. Introduction

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

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

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 [4], 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 [5] and the European Commission[6]. 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., 7–10], twitter data [e.g., 11, 12] and relational or tabular data [e.g., 13, 14]. 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.
- We create an open source evaluation data corpus.
- We provide statistics that illustrate the performance of the developed de-identification algorithm.
- We provide an open source validation algorithm and ground truth.

In Section 2, we illustrate how DDPs from different platforms can vary greatly in structure and content. In Section 3, 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 4 we describe the data used for the development and evaluation of our proposed algorithm, which is extensively discussed in Section 5. The

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

outcome of the evaluation study is presented in Section 6, followed by suggestions for future work in Section 7 and conclusions in Section 8.

2. 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 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:

- An ever changing file structure,
- both visual and textual content,
- different file formats,
- files ranging from highly structured to highly unstructured formats,
- the masking of usernames of natural persons or other users.

3. Related work

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) [15] 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 [7, 8, 16, 17]. 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 [7, 16, 17]. Scientific open-source de-identification tools are available such as DEDUCE [8] and Amnesia [18] as well as commercial tools, such as Amazon Comprehend [9] and CliniDeID [10] [19]. Most automatic de-identification approaches are constrained

		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

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 [7].

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 [20]. Anonymizing social media data is a challenging task due to their heterogeneous, highly unstructured and noisy nature [20]. Commonly used statistical disclosure control approaches [13, 14, 18, 21, 22] 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 [11, 12], 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 [12, 23]. 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 [20, 24–26]. 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 [27], and Facenet-PyTorch [28]. However, modern image recognition methods based on deep learning have demonstrated that hidden information in blurred images can be recovered [29].

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 [30]. 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.

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.

4. Data

4.1. 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.

	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

4.2. 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.

4.3. Annotating validation corpus

4.3.1. 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 3, 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.

4.3.2. 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

[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.

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

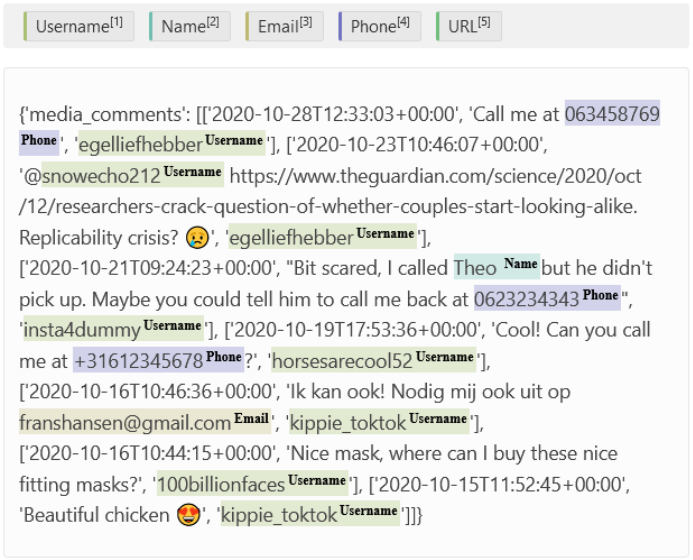

Fig. 1. An example of how labeling a comments.json file would look like in Label-Studio.

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) [31].

5. Method

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
   - De-identify PII with corresponding de-identification codes
(3) De-identify media files by detecting and blurring human faces and text

5.1. 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.

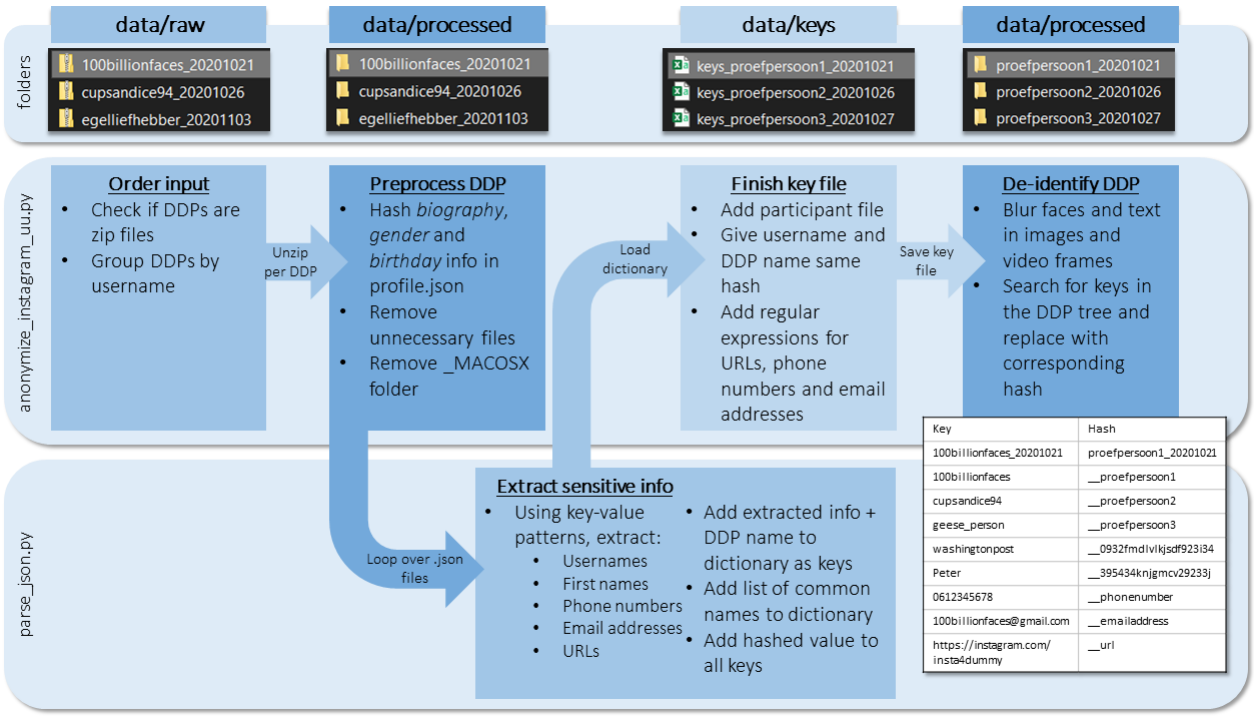

Fig. 2. The algorithm takes a zipped DDP as input. Looping over the text (.json) files, all unique instances of PII are detected in the structured part of the data using pattern- and label recognition. The extracted info, together with the most common Dutch first names and, optionally, the participant 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 algorithm will return a de-identified copy of the DDP in the output folder.

5.2. De-identify text files

5.2.1. 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 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 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.

		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.

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 [8]). Of course, it is also possible to add another list (of another country), making the algorithm applicable in multiple languages.

5.2.2. De-identifying PII in text files

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.

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.

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.

5.3. 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 [31] in Facenet Pytorch [28] and blur all occurrences using the Python Imaging Library [32]. We detect text using a pre-trained [33] EAST text detection model [34] and blur all occurrences using the Gaussian blur option provided by OpenCV [35].

5.4. 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.

5.4.1. 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.

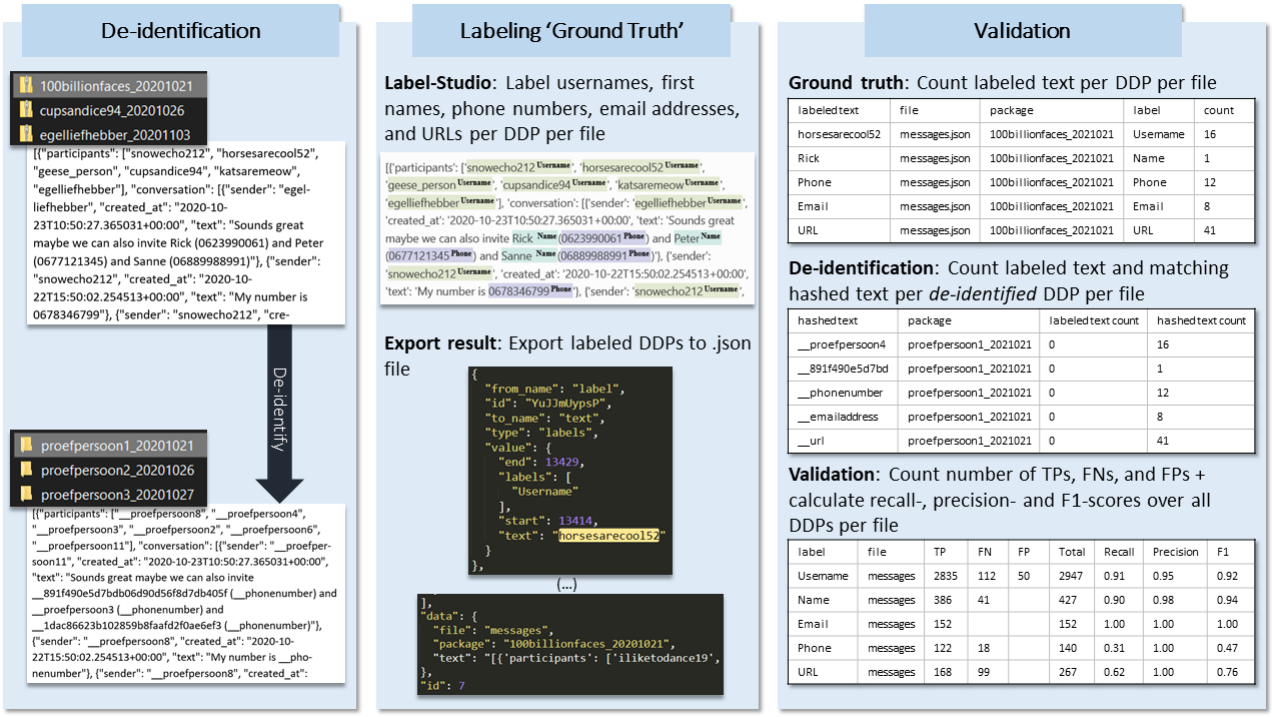

Fig. 4. The raw DDPs in which all PII categories are labeled (i.e., the ground truth) is compared with the de-identified DDPs. The algorithm counts the number of PII categories (total), correctly hashed PII (TP), falsely hashed information (FP), and unhashed PII (FN). Subsequently, a recall-, precision-,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

5.4.2. Visual content

The human rater determined for each detected face whether it was indeed de-identified by the algorithm. The definition of identifiable used (i.e., at least three out of five facial landmarks were visible [31]), 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.

5.5. Evaluation criteria

We use scikit learn to further evaluate the performance of the procedure on the different aspects [36]. First, we calculate the recall, or the sensitivity, as

$$Recall = \frac{TP}{TP + FN}. \tag{1}$$

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

$$Precision = \frac{TP}{TP + FP}. \tag{2}$$

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

A large proportion of faces on images were appropriately detected and blurred (Table 5), 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 5), 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 5 and investigating what coding decisions led to the least optimal results, improvements to the code were made.

6.2. Code adjustments

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 6).

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 7). Regarding usernames, both the number of false negatives and false positives decreased substantively.

7. Limitations and future work

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

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 5

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

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 6

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

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

	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 7

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

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 [4]. 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 [37]. 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 [29].

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.

Acknowledgements

The authors would like to thank the researchers from the Utrecht University Human Data Science Group, the Utrecht University Research Engineers and Project AWeSome for their input and comments on earlier versions of this manuscript and for their participation in the generation of the validation corpus. In addition, the authors would like to thank the reviewers for their useful comments.

Supplementary material

The de-identification algorithm is available at https://github.com/UtrechtUniversity/anonymize-ddp. The validation set containing 11 Instagram DDPs is available at https://zenodo.org/record/4472606#.YN92x-gzaUk.

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] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

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

[10] 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).

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

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

[19] 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.

[20] 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.

[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] 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.

[24] 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.

[25] 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.

[26] 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.

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

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

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

[30] 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.

[31] 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.

[32] 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.

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

[34] 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.

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

[36] 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----------.

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

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

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

[40] 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.

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

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

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

[44] 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.

[45] 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.

[46] 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.

[47] 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.

[48] 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).
[49] G. Coppersmith, M. Mitchell, C. Harman, M. Dredze and R. Leary, Deidentify Twitter, 2017. https://github.com/qntfy/deidentify_twitter.
[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.