CURLICAT

Curated Multilingual Language Resources for CEF.AT

Agreement number: INEA/CEF/ICT/A2019/1926831

Action No: 2019-EU-IA-0034

CEF programme

Deliverable 3

Anonymization

Curated Multilingual Language resources for CEF.AT

Version 1.0

2022-11-30

Document Information

Activity:	Activity 3: Anonymisation
Deliverable number:	D3
Deliverable title:	Data anonymised and their intellectual property rights clarified
Indicative submission date:	2022-06-30
Actual submission date of deliverable:	2022-11-30
Main Author(s):	Tamás Váradi, Svetla Koeva, Radovan Garabík, Andraž Repar, Bartłomiej Nitoń, Vanja Štefanec, Elena Irimia
Participants:	Bence Nyéki, László János Laki, Zijian Győző Yang, Simon Krek, Maciej Ogrodniczuk, Piotr Pęzik, Marko Tadić, Radu Ion, Vasile Păis
Version:	V1.0

History of versions

Version	Date	Status	Name of the Author (Partner)	Contributions	Description/ Approval Level
V0.1	2022-08-30	Completed	JSI		
V1.0	2022-11-30	Completed	IBL, MTANYTI, JULS SAV, JSI, RAKAI, UZ, ICS	Tamás Váradi, Svetla Koeva, Radovan Garabík, Andraž Repar, Bartłomiej Nitoń, Vanja Štefanec, Elena Irimia, Bence Nyéki, László János Laki, Zijian Győző Yang, Simon Krek, Maciej Ogrodniczuk, Piotr Pęzik, Marko Tadić, Radu Ion,	

				Vasile Păis	

EXECUTIVE SUMMARY

This activity aims at removing or anonymizing all personal and sensitive data from the language resources collected in Activities 1 and 2. By performing this, it will be ensured that the personal and sensitive data are anonymised in the textual data obtained from a wide range of sources.

An additional result of this activity is an application for anonymization of sensitive and personal data for the seven languages involved in the project (Slovenian, Croatian, Bulgarian, Romanian, Slovak, Polish and Hungarian) with language-specific anonymization modules and a common user interface.

1. Introduction

The activity includes the following tasks:

Task 1: Identification and clarification of anonymization requirements
The data collected in Activities 1 and 2 will be analysed, whereas, particular attention will be paid to the type of personal data present. Following this, the relevant anonymization method will be identified.

Task 2: Development of anonymization solutions
Based on the outcome of Task 1, various existing machine-learning approaches will be adapted and new approaches utilizing deep neural networks will be developed to provide effective anonymization solutions for the targeted languages.

Task 3: Anonymization of collected data
Using the solutions developed in Task 2, any personal and sensitive data in language resources collected in activities 1 and 2 will be anonymized to alleviate any legal concerns of the resource owners.

This activity will result in anonymized enhanced multilingual corpus of representative language data and in the anonymization solution, comprising the algorithm and user interface, for the targeted languages.

2. Conceptual background

The process of anonymisation, simply put, means removing personally identifiable information, i.e. information from which individual persons could be identified, thereby compromising or infringing their right to privacy. In the context of texts, this means replacing or removing parts of texts that contain this data.

From the point of view of automating the process of anonymising texts, it is important to distinguish between several different possible levels of anonymisation. To illustrate the difference between these, let us take the following fictional text as an example:

"On Thursday, 1 February 2020, the accused John Smith robbed Mary Johnson.
Police units caught John Smith a day later in London."

The simplest way to anonymise is to simply remove all sensitive data:

"The accused XXX robbed XXX in XXX.
XXX was caught by police units a day later in XXX."

This approach effectively removes all sensitive information, but the readability and comprehensibility of the text may suffer to the extent that it introduces ambiguity or confusion. The advantage is that this approach requires the simplest input data from a technical point of view and the simplest model structurally. If we retain a little more information and keep the type of data removed in the anonymisation process, we can obtain:

"The accused [PERSON] robbed [PERSON] on [DATE].
Police units caught [PERSON] a day later in [LOCATION]."

This ensures that sensitive data is still well protected and the text is easier to read. However, the problem of anonymisation becomes technically rather more difficult, as the automatic anonymiser now has to distinguish between different types of data. Learning such systems also requires richer input data, with different target data types labelled in the texts. Otherwise, we can use existing entity recognition models that recognise types such as: personal name, organisation, place, date and some others. The problem arises with more specific data types that are rare in general texts, such as vehicle registration numbers or bank account numbers. In addition, the text can still be confusing as all references to the same data type are replaced by an identical code. We could go one step further and disambiguate these references:

"The accused [PERSONA1] robbed [PERSON2] on [DATE1].
Police units caught [PERSONA1] a day later in [LOCATION1]."

In this form, the text is even clearer and easier to understand. However, this approach is technically the most difficult, as the automatic anonymiser has to recognise from the context when an entity is mentioned. There are approaches that solve this problem, but they are not completely reliable. For example, in the special case where two people with the same first and last name appear in a text, humans are able to tell which person is being referred to from the context, but this is a major challenge for an automatic system.

Let us also mention that, with some effort, it is potentially possible to make the display form of the text a little more attractive by replacing it with pseudonyms. For example, in the case of personal names, we can use initials, e.g. "J.S.". However, there are some potential problems with this procedure (e.g. what to do when the initials of several persons are the same) which can complicate the automatic processing, so caution is needed in the choice of method. Also, the possibilities for customised replacement depend on the anonymised text segment having a known type. If a text segment that does not correspond to any type is removed, it is of course not possible to use a specially adapted pseudonym.

As a final thought, caution is always needed when using these types of automatic systems, as they can never guarantee complete reliability. Moreover, in the case of anonymised documents, motivated individuals may be able to identify the individuals involved from contextual information, even if the documents themselves are completely anonymised.

3. CURLICAT anonymization

At the start of the project, the project partners analyzed their existing corpora and established whether additional texts from new data providers would be needed to fulfil the requirements of the CURLICAT project. For details, see reports on Activities 1 and 2. While most of the data already available to the partners could be shared in their existing form without any need for anonymization, the concern was that potential new text donors would be reluctant to share their data if it contained sensitive or personal information.

Given that specific requirements for anonymization could differ from language to language and because language models and other resources may not be available for all languages involved in the project, it was decided that each partner would develop their own language-specific solution and that these solutions would be integrated into a single interface by JSI. The Polish and Croatian partners determined that their texts would not need any anonymization, so JSI also developed a general approach for these two languages and Slovenian, while the other languages (Slovak, Hungarian, Romanian, Bulgarian) were taken care of by the respective project partners.

Each language-specific approach was dockerized and a common web application which runs as a separate docker container and communicates with the containers of anonymisation models was developed by JSI. The application is available at http://ircai.ijs.si:7000/.

3.1. Requirements for the anonymisation model

Each anonymisation model is dockerized in its own container. When started, it listens on a specific (configurable) port, accepting POST requests and returning responses in the format described below.

3.2. API Calls

Request

The POST request (from web application to each anonymisation model) contains the data in JSON format with the following structure:

```
{
  "text": "Mr. John White from London.",
  "format": "text"
}
```

Description of the fields:

FIELD	EXPLANATION
text	Text that should be anonymized. Can be in either *plain text* or *CONLL* format.
format	Specifies the text format. Set to “text” for *plain text* format and to “conll” for *CONLL* format.

Example of sending such request with curl command:

```
curl -X 'POST' \
  'http://model_en:5000/anonymize \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "text": "Mr. John White from London.",
  "format": "text"
}'
```

Response

The response (from anonymisation model back to the web application) contains the data in JSON format with the following structure:

```
{
  "original_text": "Mr. George White from London.",
  "anonymized_text": "Mr. Peter Black from New York.",
  "format": "text"
}
```

The description of fields:

FIELD	**EXPLANATION**
original_text	Text that was sent for anonymisation. Can be in either *plain text* or *CONLL* format.
anonymized_text	Anonymized version of original_text. Should be in the same format as anonymized_text.
format	Specifies the format of original_text and anonymized_text. Set to “text” for *plain text* format and to “conll” for *CONLL* format.

Note that the web application does not do any kind of annotation or format conversion (CONLL to plain text or vice-versa) but simply forwards the text from the user request to the selected anonymisation model. Anonymisation models are required to support the CONLL format used in the Curlicat project. The resulting anonymised version of the text, where the entities are replaced or obfuscated in some other way, should also be in the valid CONLL format when format: "conll" is used.

4. Bulgarian approach

The anonymization for Bulgarian is based on the anonymization models provided by the MAPA project (https://gitlab.com/MAPA-EU-Project/mapa_project) and regular expressions working with dictionaries of named entities and their triggers. All triggers and named entities are stored in a SQLite3 database. The following applications have been used:

- MAPA v2.3 for text file analysis

– php-7 as a web server to handle API requests, pipeline workflow, and regular expressions

– SQLite3 is used for dictionary storage

The identified named entities are replaced with pseudonyms. In index.php and anonymization.php, some predefined lists of names are coded and used for replacements while attempting to preserve original word form; for example, if a segment is "family name" and the preceding segment is "given name -- male, it is replaced with a random male family name. Replacement rules obscure the names of persons, organizations, and locations and obscure identification numbers. IBANs, VAT numbers, URLs, IPs, phone numbers, emails, dates, MAC addresses, GPS locations, etc.

5. Hungarian approach

The Hungarian approach is a pipeline that implements a pseudo-anonymization tool. The pipeline integrates different named entity recognition, morphological parsing and morphological generation modules. Our tool does not simply recognize and remove/hide the named entities, but it also can replace the found name with another name consistently in a given document. In our pipeline we have five main modules and process steps:

1. **Preprocessing**: For preprocessing, HuSpaCy was used. The input text is splitted into sentences and a tokenization process is applied.
2. **Morphological analyzer**: HuSpaCy and emMorph modules were integrated. Morphological analysis can be performed with the HuSpaCy tool in UD format. The emMorph is an Hungarian morphological analyzer that uses Humor unification morphology.
3. **NER**: For named entity recognition, a fine-tuned huBERT model was used.
4. **Name database**: The names were collected from the official list of Hungarian surnames that are recognised.
5. **Morphology generator**: Two neural-based Hungarian morphology generators were used. One emMorph and one UD morphology generator. The generator models were trained with the Marian NMT machine translation system.

An example:

- input text: Édes, Drága, áldott **Adélom**.
- output text: Édes, Drága, áldott **Darlám**.

The pipeline (including modules and models) is built into a Docker container, which is available a the following link: https://git.nlp.nytud.hu/CURLICAT/pseudo-anonimization (Authorization will be available soon.)

https://curlicat-project.eu/assets/downloads/CURLICAT_D6.1._Dissemination_plan_v1.0.pdf

The input and output format is the same as described in Section 3.2.

6. Romanian approach

Anonymization of the Romanian corpus is performed following a named entity recognition module. Identified named entities are then replaced with pseudonyms. These are generated randomly, while trying to preserve the initial word form, including characteristics such as suffixes or prefixes. The algorithm is described in detail in (Păiș et al., 2021) and is available online in the RELATE platform (Păiș et al., 2020).

The anonymization implementation is further available in our GitHub repository: https://github.com/racai-ai/ROAnonymization_CURLICAT. The implementation can be assembled into a Docker container exposing a JSON REST API.

7. Slovak approach

Anonymization in Slovak model is performed for nouns that are marked as names of persons[1] by the Named Entity Recogniser.

The anonymization works by replacing a triplet (*wordform, lemma, tag*) by a substitute (*wordform_anon, lemma_anon, tag*), where both *lemma* and *wordform* are replaced by the most frequent (frequencies are obtained from the corpus *prim-9.0-juls-sane*[2]) substitute lemma of the same inflectional paradigm (consequently, also of the same grammatical gender) defined as the same sequence of Levenshtein edit operations transforming the lemma to the inflected wordform, based on the database used by the dictionary of noun paradigms (Garabík et al. 2016). The substitution is performed only if there are at least three different titlecase nouns in the set of the same inflectional paradigms; the substitute lemma and wordform are marked by preceding text string 'XXX_'. If the substitution cannot be performed (either the lemma is unknown to the database of inflectional paradigms, or the inflected wordform is inconsistent with the tag), the wordform and lemma are substituted by the string 'XXXX' followed by the last character[3] of the wordform and the lemma, respectively.

Example:
Lexeme *Ivan* in dative singular:
Ivanovi Ivan SSms3 → XXX_Jánovi XXX_Ján SSms3

Lexeme *Prešeren* (surname not present in the Slovak morphological database, but still appearing in general Slovak texts, although rare) in dative singular:
Prešerenovi Prešeren SSms3 → XXXXi XXXXn SSms3

Given the necessity to include lemmatization and full MSD on the input, the anonymization is implemented by performing the initial part of CURLICAT text processing pipeline (tokenization, lemmatization, MSD, NER) and then applying the anonymization step.

The Slovak anonymization docker follows the general API described in the section *2. CURLICAT anonymization* of this document. The docker does not recognize the 'Expect' http header, which has to be blanked explicitly if using *curl* for connection and sending data of nontrivial size.

If using the *conll* format, the input is expected to be lemmatized, MSD tagged and annotated for named entities according to the CURLICAT specifications.

[1] This is a configurable parameter that can be set to include e.g. geographical locations as well.

[2] https://korpus.sk/korpusy-a-databazy/korpusy-snk/struktura-korpusu-prim-9-0/

[3] The digraphs *ch*, *dz* and *dž* are considered one character each, following the existing custom in Slovak lexicography.

8. Slovenian, Croatian and Polish approach

The Slovenian, Croatian and Polish anonymization approach uses two components, the anonymization model (anonymizer) and entity replacement model (replacer).

First, the entities in the input text that should be anonymized are detected by the anonymizer. For example, given the sentence *"Janez Novak iz Ljubljane"*, the anonymizer outputs *"~~Janez Novak~~ iz ~~Ljubljane~~"* Then, each of the detected entities is replaced with the synonym using the replacer. The replacement is done on a single word at the time by masking it and using the replacer to find the alternative. For example, the sentence *"<MASK> Novak iz Ljubljane"* is transformed to *"Luka Novak iz Ljubljane"* in the first step, while in the second step, the sentence *"Luka <MASK> iz Ljubljane"* is transformed to *"Luka Horvat iz Ljubljane"*.
To further increase the quality of the text generated by such transformations, we use the additional annotation component and require that the linguistic annotations (such as part-of-speech and named entity tags) of the original sentence match the annotations of its transformation.

The implementation details of each components are the following:

Anonymizer
The anonymization model is based on the named entity recognition model SloNER which was developed in the RSDO project. The SloNER model is based on multilingual BERT (https://huggingface.co/bert-base-multilingual-cased) and was trained on the publicly available annotated data in Slovene language containing different types of named entities, such as personal names and organizations. the model assigns the BIO label to each word in the input text. For example, the sentence *"Janez Novak iz Ljubljane"* is tagged as [B-PER, I-PER, O, B-LOC], where PER refers to the personal name and LOC refers to location.

The anonymizer is initialized with the weight from the SloNER model and is further trained on the semi-automatically labeled documents from the legal domain. Each word in the document is labeled with either anonymized or non-anonymized. The model assigns binary (I or O) label to each word in the input text. For example, the sentence *"Janez Novak iz Ljubljane"* is tagged as [I, I, O, I], where I indicated that the word is anonymized.

Replacer
The replacer model is a BERT model trained on Slovenian documents called SloBERTa (https://www.clarin.si/repository/xmlui/handle/11356/1397). We use the masked language modelling capability of the pretrained BERT model to find possible replacements for each word. The model produces 40 alternatives and we select the single one using the annotation component.

Annotator

We use the Classla annotation tool (https://github.com/clarinsi/classla) which is a fork of Stanza tool from Stanford (https://github.com/stanfordnlp/stanza) for Processing Slovenian, Croatian, Serbian, Macedonian and Bulgarian. We use the tool to ensure that each sentence obtained after the word replacement retains the same structure in terms of POS and NER tags as the original sentence.

The tool is distributed as a docker image and follows the API described in section 2.

8.1 Polish

The Polish implementation uses the same approach as described above for Slovenian, except that the replacer model for Polish uses herbert-base-cased checkpoint of the BERT model available at: https://huggingface.co/allegro/herbert-base-cased. The annotator is implemented using Spacy (pl_core_news_md model).

The tool is distributed as a docker image and follows the API described in section 2.

8.2 Croatian

The Croatian implementation uses the same approach as described above for Slovenian, expect that the replacer mode for Croatian uses Andrija/SRoBERTa-XL checkpoint of the BERT model available at: https://huggingface.co/Andrija/SRoBERTa-XL The annotator is implemented as in Slovenian approach using Classla.

The tool is distributed as a docker image and follows the API described in section 2.

9. Summary of results and conclusion

This activity resulted in the development of an anonymization and replacement solution for the 7 languages (Bulgarian, Croatian, Hungarian, Romanian, Slovenian, Polish, Slovak) involved in the project. Given the partners' experience with text acquisition for national language corpora, it was our belief that anonymisation could potentially play an important role in convincing potential data providers to more willingly share their data, which was the main reason behind the inclusion of this activity in the project. Additional discussions with data providers after the start of the project were also encouraging with several providers expressing interest in the possibility of provided anonymized data.

However, the current-state-of-the-art approaches using neural NER models and transformer-based replacement do not perform completely without errors and even the most advanced anonymization algorithm will introduce a certain amount of noise into the source texts. Coupled with the fact that the texts we managed to acquire were already available in the public domain or under permissive licenses, and to provide data of the highest possible quality, we did not anonymize the corpora (with the exception of Romanian).

The final result of the activity is thus the anonymization tool, available for demonstration purposes at http://ircai.ijs.si:7000/, and its code (to be made public in the Technical Report).

10. Bibliographical references

Garabík, R., Karčová, A., Šimková, M., Brída, R., Žáková, A. (2016). Skloňovanie podstatných mien v slovenčine s korpusovými príkladmi. Bratislava, Vydavateľstvo Mikula.

Kilgarriff A. (2009). Simple maths for keywords. In Proceedings of Corpus Linguistics Conference CL2009, Mahlberg, M., González-Díaz, V. & Smith, C. (eds.), University of Liverpool, UK.

Koeva S., Stoyanova I., Leseva S., Dekova R., Dimitrova T., Tarpomanova E. (2012). The Bulgarian National Corpus: Theory and Practice in Corpus Design. Journal of Language Modelling (1), pp. 65–110. https://doi.org/10.15398/jlm.v0i1.33

Logar, N., Grčar, M., Brakus, M., Erjavec, T., Arhar Holdt, Š., Krek, S. (2012). *Korpusi slovenskega jezika Gigafida, KRES, in ccKRES: gradnja, vsebina, uporaba*. Ljubljana: Trojina, zavod za uporabno slovenistiko: Fakulteta za družbene vede.

Mititelu V. B., Tufiș D., Irimia E., Păiș V., Ion R., Diewald N., Mitrofan M., Onofrei M. (2019). Little Strokes Fell Great Oaks. Creating CoRoLa, The Reference Corpus of Contemporary Romanian. Revue Roumaine de Linguistique, No./Issue 3.

Oravecz C., Váradi T., Sass B. (2014). The Hungarian Gigaword Corpus. In Proceedings of LREC 2014.

Păiș, V., Tufiș, D., Ion, R. (2020). A Processing Platform Relating Data and Tools for Romanian Language. In Proceedings of The 12th Language Resources and Evaluation Conference. European Language Resources Association, Marseille, France, pp. 81-88.

Păiș, V., Irimia, E., Ion, R., Tufiș, D., Mitrofan, M., Barbu Mititelu, V., Avram, A.M., Curea, E. (2021). Romanian text anonymization experiments from the CURLICAT project. In The 16th International Conference on Linguistic Resources and Tools for Natural Language Processing. pp. 165--178.

Przepiórkowski A., Bańko M., Górski R. L., Lewandowska-Tomaszczyk B. (2012). Narodowy Korpus Języka Polskiego. PWN Scientific Publishers.

Tadić, M., (2009) New Version of the Croatian National Corpus. In Hlaváčková D., Horák A., Osolsobě K., Rychlý P. (eds.) After Half a Century of Slavonic Natural Language Processing, pp. 221-228, Tribun EU, Brno.

Ulčar, Matej and Robnik-Šikonja, Marko, 2021, Slovenian RoBERTa contextual embeddings model: SloBERTa 2.0, Slovenian language resource repository CLARIN.SI, ISSN 2820-4042, http://hdl.handle.net/11356/1397.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton and Christopher D. Manning. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. In Association for Computational Linguistics (ACL) System Demonstrations. 2020.