sciendo Journal of Electrical Engineering, Vol. 76, No. 3, 2025, pp. 275-283

Unstructured documents processing through their anonymization

Peter Kvasnica

In Slovakia, a large number of texts on decisions are published by the courts on the MSSR website. All decisions that are published must be anonymous, without all personal and other data (personal and sensitive data that can identify individual entities in the proceedings, reveal their private data, etc.) to the extent described by the decree. Anonymization of court decision data is carried out according to an instruction given in the Court Management application, which supports only basic anonymization functions. Such an anonymization method is time- and human-resource-intensive, and at the same time, it is susceptible to the leakage of sensitive data. The text processing is based on machine learning methods using the principles of user-friendly visualization to accelerate anonymization and minimize the risk of leaking sensitive data. Achieving and improving results in practice the method of recognizing named entities based on the processing of text documents, dictionaries, rules and a trained statistical model on manually annotated data is used. The article describes the basic techniques of the anonymization used in practice, that are machine learning and removing information. Our attention is given to text documents processing, tools and principles for creating a data set of annotated data and evaluating the success of anonymization. The iteratively improving learning model is dominant in that it enables the use of knowledge as well as annotated data.

Kewords: data anonymization, named entity recognition, dataset information extraction, artificial intelligence, machine learning

1 Introduction

In Slovakia, around 28,000 decisions are made and published by the courts on the MSSR website per month [1]. Most of the decisions are published on the web pursuant to the paragraph 82 letter and Act No. 757/2004 Coll. on Courts, and pursuant to the Decree on Publication of Court Decisions No. 482/2011 Coll. [2]. Some decisions that are not published are available in the context of the Act on Free Access to Information No. 211/2000 Coll. The request for the information is in an anonymized form and is additionally made anonymous and provided to the applicant [3].

Court decisions are publicly available according to the so-called document “Anonymization of court decision data” pursuant to instruction issued on 23/2011 by the Slovak Ministry of Justice. Court officials use the anonymization of court decisions in the Court Management module. This allows court decisions to be anonymized in two methods. The first method includes text anonymization when highlighted parts of texts are automatically anonymized. The other method includes the anonymization window, in which it is possible to jump between words that start with a capital letter or number. When jumping, it is possible to choose whether to anonymize the given entity (word/date/number) or not. Such anonymization is lengthy and the existing tool does not significantly facilitate the work of the user in identifying which entities need to be anonymized and which do not.

The above mentioned anonymization module is a type of application available in Court Management, which provides a user interface to make a decision. When texts of decisions are designed and completed, the highlighted parts of the text will be replaced by wildcards in accordance with the decree. Subsequently, it is possible to anonymize the decision texts using the anonymization tool, which forms a part of the Court Management application. Keyboard shortcuts or the keys to jump between words of anonymized texts are used for the described activity. Using a keyboard shortcut or a key, the selected word or numerical expression can be anonymized. The previous system does not support user decision-making and is user-unfriendly due to the number of necessary actions and the time spent anonymizing one document [4].

The anonymization service has a user friendly interface that offers the necessary functionality for working with an anonymized document. The service evaluates its success and in future the required goal of the service is fully automatic functionality of the anonymization process without the need for user intervention. The service allows you to add new words to dictionaries, correct the proposed anonymization of the entity, or mark the type of entity in the text, thereby ensuring the improvement of the quality of results (learning) in subsequent anonymization processes.

Department of Information and Project Control, Ministry of Justice of Slovak Republic, Račianska 71, Bratislava, Slovak Republic
peter.kvasnica@justice.sk

https://doi.org/10.2478/jee-2025-0028, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

Substantial improvement of the work in the anonymization process is brought about by the presented method using the processing of text documents and information extraction. It is designed and implemented as an extension of the functionality of the Integration Platform (IP) module in the department's application architecture and security infrastructure information system (IS BAI) with a universal service for anonymizing court decisions. This service provides a wider range of functionality and at the same time speeds up and simplifies the anonymization process for various information systems and modules within the department [4]. The proposed method focuses on the processing of text documents and information extraction, it consists of three tasks linked to it and supporting processes: Rapid Annotation Tool (RAT), Named Entity Recognition (NER) and Hybrid method of manual anonymization.

The solution significantly saves the time of the worker performing the anonymization. The application is characterized by a better way of guidance in the document, while sensitive and non-sensitive data are highlighted and differentiated in color from the point of view of anonymization. The document prepared in this way is anonymized using an artificial intelligence (AI) algorithm [4].

An important fact of the proposed method is the reduction of sensitive data leakage in published documents. A prerequisite for preventing the leakage of sensitive information from documents is a high-quality initial learning of the artificial intelligence anonymization module, setting rules and creating dictionaries. However, the initial model prepared in this way is subject to constant learning and improvement based on corrections. Corrections of anonymized documents are performed by users who independently check the results of anonymized documents from the learned model [5]. In order for the mentioned procedure to be able to improve in the detection of data that needs to be anonymized, it is necessary to extract structured data from documents for further use and search.

The Slovak language belongs to the branch of West Slavic languages (Slovak, Czech, Polish and Lusatian Serbian) [6]. The rules of morphology are quite complicated and it is not always easy to determine the root of a word.

In lemmatization, the biggest problem is with the form homonyms, e.g., the word “is” may be the present tense of the verbs to be or to eat. Therefore, even if a dictionary is available, we cannot determine a clear lemma of a stand-alone word. When making a decision, we must use the context of the word, its position in the sentence, the tense or case in which it is found, and it must also fit into the sentence meaningfully. When searching for information, this problem is not so striking, but when extracting information, it can cause big differences.

The advantage of Slovak over Germanic languages when extracting information is the use of suffixes for surnames of persons, and thus it is easier to identify the gender of a person by name and surname. In Slovak, unlike Germanic languages, the position of words does not play such a big role in identifying the part of speech [7].

2 Text documents processing

For the automatic processing of information and subsequent derivation or their automated use, we need to extract data from documents, and this is what information extraction is for. Extraction is performed using a hybrid method of named entity recognition, which combines the use of manually created rules in the form of token regular expressions with the use of Conditional Random Fields (CRF) statistical model [8]. We increase the quality by automatically extracting references to legislative regulations in the document, the number of which is continuously increasing. Applying this method using the mentioned procedure is the subject of the presented article.

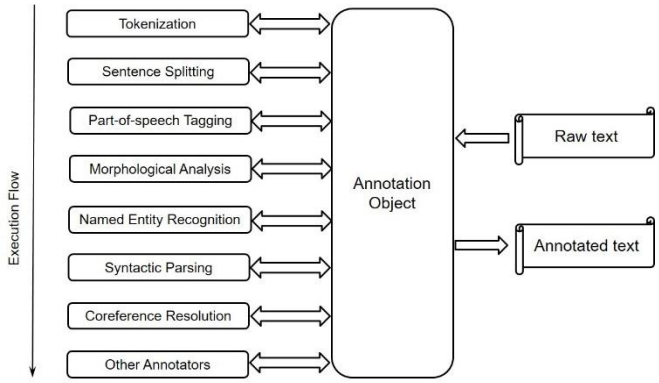

Fig. 1. Scheme of document anonymization subprocesses

In Figure 1, in the left part, horizontally placed separate subprocesses are listed which are performed in sequence from top to bottom when anonymizing a document. It is important to realize that after each separate sub-process data is transferred to the anode object. Each sub-process represents one tool when working with spoken language and has defined what annotations it produces and what it requires from previous annotators.

Information extraction is a general term and a broad field covering various problems. To solve more complicated problems, such as recognition of named entities, it is necessary to use a whole range of tools for working with natural language. Information extraction is a field

that solves many problems and is divided into several tasks. At the Message Understanding Conferences (MUC), five basic tasks of IE (from the English information extraction) were defined.

- **Named entity recognition** (NER). It extracts and classifies persons, cities, temporal expressions and the like. An example of a named entity is *"Faculty of Electrical Engineering and Informatics"*.
- **Session extraction** (Coreference resolution, CO). Finding and identifying relationships between entities extracted using NER. An example is the relationship between *"FEI"* and *"Faculty of Electrical Engineering and Informatics"*, where it is the same entity.
- **Construction of template elements** (Template elements, TE). Adds descriptive information to entities from NER with the help of CO.
- **Construction of template relations** (Template relations, TR). Finding relationships between entities and their properties identified in TE.
- **Production of scenario templates** (Scenario template, ST). It enables the fitting of results from TE and TR to a specific problem domain.

For the needs of solving more complex problems, toolkits for working with spoken language, such as CoreNLP, OpenNLP, Natural Language Toolkit (NLTK) or GATE, were created.

The proposed solution, using analysis and information extraction from the text, obtains named entities from the text and subsequently decides whether the data is of a personal nature, subject to personal data protection and therefore needs to be anonymized [4]. Methods for working with text in natural language, based on the use of entity extraction, are domain dependent. This fact means that the methods use extensive general laws about language-ontologies, domain vocabularies and are specialized for a specific domain.

For our solution, we present the following test entities, which we will extract from documents of court decisions: Test set of entities for the NER method (Court, ID number, Person, Address, Date of birth, Organization) on (Physical person, Legal person, Follow-up file, Cadastral data, etc.). The entity Natural person contains: Natural person, Date of birth, Address (residence), Social security number, OP number, Travel document number, Telephone and Date of death. The File Data entity contains: Judge, Court File tag, URL to law, Legislation [6].

2.1 Procedures used to test entities

Process results from user testing of text document processing and embedded suggestions are dominant in this phase of training set preparation. The knowledge provides mechanisms for preprocessing, linguistic analysis and indexing of large collections of text documents, as well as for creating, testing and evaluating models of data mining in texts by machine learning algorithms.

Testing is done in several ways for training artificial intelligence. Each method has advantages and disadvantages for practical use and training, which are determined by real conditions [8].

When anonymizing the person entity, we used the following procedures:

- the first procedure uses known persons from the file of the court management (SM) and finds them in the identified persons and, based on the information from the SM, they are marked for anonymization,
- the second procedure finds a person known from the registers of judges, lawyers, notaries, ... and marks them for anonymization, if necessary, based on their symptoms,
- the third procedure, based on the trained set and the context, identifies persons who need to be anonymized and who should not be anonymized.

The described procedure also anonymizes entities: Address, Date of Birth, Organization and Legislation.

3 Machine learning in the processing of text documents

In this part, we will present the processing of text documents from various sources of information systems. This forms the core of this work. When the problem of processing text documents is raised in connection with machine learning, it is usually the categorization of text documents. The goal of text categorization is to find an approximation of the unknown function $\Phi: D \times C \rightarrow \{true, false\}$, where D is a set of documents and $C = \{c_1, ..., c_{|C|}\}$ is a set of predefined categories. The value of the function Φ for the pair $\langle d_i, c_j \rangle$ is equal to the value *true* (or *false*) if the document d_i belongs (or does not belong) to the category c_j. In fact, the classification algorithm does not learn the function Φ from the training examples – documents, but an approximation of this function $\bar{\Phi}$ which we call a classifier [9].

Definition of text categorization is based on categories are only nominal labels and there is no knowledge (either declarative or procedural) of their meaning and categorization is based on the extraction of knowledge from the text of documents.

Depending on the specific application, it is possible to limit the number of categories for which the function Φ has the value *true* for a given document d_i. If a document d_i can be classified into exactly one class $c \in C$, then C represents a set of disjoint classes. If every document can be classified into $k = 0, ..., |C|$ classes from the set C, then it is a multiclass classification and C represents a set of overlapping classes [9].

Binary classification represents a special case of multiclass classification, namely classification into two classes. In principle, the binary classification algorithm can also be used for multiclass classification. If the classes are independent of each other (i.e., for each pair of classes c_j, c_k and $j \neq k$, the value $\Phi(d_i, c_j)$ is independent of the value $\Phi(d_i, c_k)$, the multiclass classification problem can be decomposed into $|C|$ independent binary classification classes $\{c_i, c_i^-\}$ for $i = 0, ...,|C|$.

3.1 Representing text documents

For the successful processing of text documents, an appropriate choice of document representation is necessary. The classic approach defines a document with a set of keywords – terms. Then the real document can be described using a vector of weights, which express the importance of each term of the set of key words for the definition of the given document. We also call this vector the lexical profile of the document. Thus, any document is represented by vector (lexical profile) $\boldsymbol{w}_i = (w_{i1}, w_{i2}, ..., w_{ij}, ..., w_{in})$ in n-dimensional space R^n, where w_{ij} is a weight expressing the importance of the j-th term in the i-th document, the value of which is non-zero if the j-th term occurs in the i-th document [9]. Among the basic models of text document representation are: Boolean, probabilistic and vector model.

The Boolean model reduces the general representation model to just using the binary values of a vector of weights. The elements of the weight vector that define the document can only take the values "1" or "0". The model therefore takes into account only the occurrence, or the absence of the given term in the document. The advantage is simplicity, but on the other hand, a rather relevant disadvantage is that information about the frequency of occurrence of a given term in a document is not considered.

Probabilistic model. In a situation where a corpus of documents is available in which a specific document needs to be searched, it is appropriate to use a probabilistic model. This model will serve well the task of sorting the found documents according to the probability of relevance of the keyword to the given document, i.e., the demand for the given document. Corpus documents, or document – query are represented by binary vectors $\boldsymbol{w}_i$ or $\boldsymbol{q}$. Similarity is defined as [10]

$$\mathrm{sim}(\boldsymbol{w}_j, \boldsymbol{q}) = \frac{P(R|d_j)}{P(\bar{R}|d_j)} \tag{1}$$

where $P(R|d_j)$ is the probability that document d_j is relevant to the query. Furthermore, R is a set of relevant and $\bar{R}$ a set of irrelevant documents with respect to the query. Using Bayes' theorem under the assumption of linear independence of attributes, it is possible to derive

$$\mathrm{sim}(\boldsymbol{w}_j, \boldsymbol{q}) = \frac{P(R|\,)}{P(\bar{R}|\,)} \prod_{i=1}^{n} \frac{P(t_i|R)}{P(t_i|\bar{R})} \tag{2}$$

where t_i is the i-th term from the set of all n terms used in the model. $P(t_i|R)$ is the probability of occurrence of term t_i in R for query q and $P(R)$ is the probability of document relevance (it is a constant for each document) [10].

Since the set R is not known at the beginning of the process, the following initialization is necessary

$$P(t_i|R) = 0.5 \tag{3}$$

$$P(t_i|\bar{R}) = \frac{n_i}{N} \tag{4}$$

where n_i is the number of documents containing term t_i and N is the total number of documents in the corpus. In this step, the model returns an initial set of relevant documents, which is further refined. Let V be the set of documents returned in iteration $(i - 1)$ and V_i be the set of returned documents containing term t_i. Then the initialization in step i will be as follows

$$P(t_i|R) = \frac{|V_i|}{|V|} \tag{5}$$

$$P(t_i|\bar{R}) = \frac{n_i - |V_i|}{N - |V|} \tag{6}$$

The disadvantage of this model is the assumption of attribute independence and the binary representation of document and query vectors. More detailed information about this model can be found in [10].

Vector model. The Vector Space Model (VSM) is the most commonly used representation of text documents [11]. The tag space for this representation is constructed based on a set of words, with each tag corresponding to a text unit in the document. A word, phrase, or another lexical unit. The given model is based only on the statistical characteristics of the document, in particular on the probability distribution of individual words extracted from the documents. The representation of the document is an n-dimensional vector

$\boldsymbol{w}_i = (w_{i1}, w_{i2}, ..., w_{in})$, where w_{ij} is a functional value expressing the weight, or the importance of the j-th word in the i-th document d_i and n is the dimension of the index set of words, i.e., those words that make up the symptoms of the description of individual documents. VSM eliminates the shortcomings of the Boolean model in that the individual weights reflect not only the occurrence of the relevant term (Boolean 0 or 1) but also describe the number of occurrences of the relevant term in the document, or its weight, i.e., its importance or significance. The vector model of the document does not take into account the structure of the document, the contextual and even the semantic information about the given term, as it is based only on the statistical characteristics of the document [11].

A set of m documents (corpus) can be described by matrix $\boldsymbol{A}$ of size $m \times n$

$$\boldsymbol{A} = \begin{matrix} d_1 \\ d_2 \\ \vdots \\ d_i \\ \vdots \\ d_m \end{matrix} \begin{matrix} = \\ \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} w_{11} & w_{12} & \cdots & w_{1j} & \cdots & w_{1n} \\ w_{21} & w_{22} & \cdots & w_{2j} & \cdots & w_{2n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ w_{i1} & w_{i2} & \cdots & w_{ij} & \cdots & w_{in} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ w_{m1} & w_{m2} & \cdots & w_{mj} & \cdots & w_{mn} \end{matrix} \quad (7)$$

If it is true that $F = \boldsymbol{A}^T$, then $w_{ij}=F(d_i, t_j)$ holds for the weight w_{ij} of term t_j in document d_i. We call function F a weight function. The specific definition of this function represents different ways of weighting the terms. According to [12], the weighting is divided into local weighting, i.e., weighting based on the number of occurrences of terms in the document itself $L(d_i, t_j)$ and global weighting $G(t_j)$ expressing the number of occurrences of the term in the entire corpus. The weighted frequency of the term t_j in the document d_i is the product of the local weight and the global weight $w_{ij} = L(d_i, t_j)G(t_j)$.

In the experiments presented in this article, a vector model of the representation of text documents was used.

3.2 Machine learning method

The essence of the method is based on text extraction for the anonymization process, where the user does not have to confirm the highlighted entity, but can agree to the anonymization proposal at once, or correct specific entities to the correct form.

In order to recognize the composite entity person, the document used the rules in combinations of the first name, last name and titles from the dictionary. The other entities mentioned in the previous text were recognized by a similar principle.

Machine learning is part of artificial intelligence that deals with methods and algorithms that allow a program to learn and then respond adequately to various input values without being explicitly programmed to them. Machine learning algorithms use elements of mathematical statistics, methods of statistical analysis and in-depth data analysis – English Data mining [13, 14]. Machine learning takes place over the concept space. The space of concepts is sometimes also called the space of solutions (a concept represents a solution to a task: learn a concept). This space of concepts is usually arranged, e.g., according to generality.

Suppose we have a group of objects (entities) described using five properties, i.e., attributes: person, address, date of birth, legislation and organization. These objects are represented using the predicate anon (anonymize, person, address, date of birth, legislation organization). The space of concepts for the given objects arranged according to generality is shown in Fig. 2.

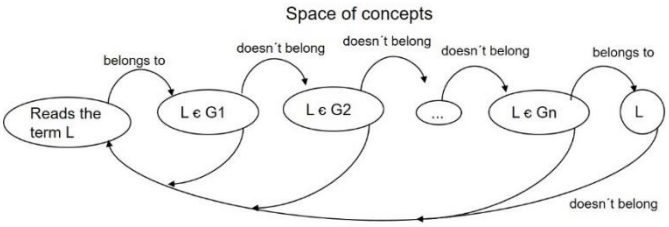

Fig. 2. Text term space for anonymization objects

After executing a sequence of training examples from the set $\boldsymbol{X}$ belonging to the target concept from the set $\boldsymbol{C}$, the learning algorithm L must output a hypothesis h from set $\boldsymbol{H}$, which is an estimate of $\boldsymbol{C}$, where [13]:

- $\boldsymbol{X}$ is a set of training examples over which the objective function is defined,
- $\boldsymbol{C}$ is a set of target terms,
- $\boldsymbol{D}$ is the probability distribution on the basis of which the training examples are selected for the training set,
- $\boldsymbol{H}$ is the set of possible hypotheses that the learning algorithm L considers when trying to learn the target concept $\boldsymbol{C}$.

3.3 Entity annotation evaluation

Entity annotation is crucial for training effective machine learning models, focus challenged the like subjectivity and handling complex text. In the machine learning landscape, entity annotation – the process of identifying and labeling key elements with text – is critical for training high performance models. This makes it an essential step in leveraging the power of unstructured testured information.

Manual annotation of court decisions was performed by one annotator at the court. The annotated decision was thus anonymized. Subsequently, the anonymized court decision was annotated by the second and third

annotator [15]. After these stages of annotation of the court decision, processing was carried out by one decision maker, who annotated 296 documents in the above manner. Finally, the results were compared with the results of the automatic linguistic method (NER), which revealed several smaller problems [16].

Text decisions have been annotated in 1,480 samples of documents (court decisions). The number of tokens is listed with each anonymized entity in parentheses. A token is a string of consecutive characters, separated by complex rules – but usually by a space or other blank character or punctuation. Data on the number of tokens when anonymizing a sample of documents are in % for anonymized entities listed in the following table.

Table 1. Results of manual data anonymization with two annotators in machine learning

Method name		Person (6776)	Address (5381)	Date of birth (464)	Legislation (44022)	Organization (7772)
Parameter/ Metric	2nd Annotator					
Accuracy		98.83	98.76	98.93	99.37	94.11
Coverage		97.24	93.35	99.57	97.50	88.78
F-score		98.03	95.98	99.25	98.42	91.37
Parameter/ Metric	3rd Annotator					
Accuracy		98.92	98.83	99.78	99.81	93.75
Coverage		99.07	96.00	99.35	99.08	92.72
F-score		99.00	97.40	99.57	99.44	93.23

Evaluation of manual anonymization performed by two annotators was performed against the gold dataset. This "benchmark" dataset was created by combining data obtained from entity anonymization by the referee and linguistic NER. A comparison of the data in Tab. 1 shows that the second annotator achieved worse/weaker coverage for the address entity than the third annotator (93.35% versus 96.00%). Similarly, the second annotator achieved a worse result than the third annotator in the coverage of the organization entity (88.78% versus 92.72%). On the contrary, the second annotator achieved slightly better results than the third annotator in the coverage of the date of birth entity and in the accuracy of the organization entity.

4 Document representation and information extraction

The processing of text documents must be preceeded by preprocessing of text documents, in order to improve the anonymization of entities. Preprocessing is a process in which the document is effectively transformed into a suitable form according to the chosen type of representation (for example, into the form of an indexed document). An indexed document can be created by manual indexing or automatic indexing [6].

CoreNLP is a product developed at Stanford University in California. It is a set of tools for working with natural language. Individual tools can be run independently or they can be run sequentially one after the other in chained text processing, which we successfully used when training datasets. For most tools, the so-called annotation inputs are available [17]. Such a solution enables easy addition of own annotators, as well as integration of existing ones.

CoreNLP core tools include:

- TokenizerAnnotator – Tokenizer with choice of implementation. The basic implementation is the PTBTokenizer, which is defined by rules in the jflex12 format.
- WordsToSentencesAnnotator – Linguistic sentence separator based on tokens identified by TokenizerAnnotator.
- POSTaggerAnnotator – Statistical POS tagger. It uses the maximum entropy model.
- MorphaAnnotator – Linguistic lemmatizer of the English language. It uses defined rules and a dictionnary in jflex format.
- TokensRegexNERAnnotator/RegexNERAnnotator/ NERCombinerAnnotator – Linguistic and statistical NER. Linguistic uses a custom language similar to regular expressions and can be used for sequences. Statistical NER uses the CRF model.

- ParserAnnotator – Parser and dependency annotator. It contains various implementations at the level of statistical model, linguistic model up to neural networks.

Each annotator represents one tool for working with natural language and has defined what annotations it produces and what it requires from previous annotators, as shown in Fig. 1.

4.1 Principle of documents data extraction

Document data extraction is the process of identifying, retrieving, and structuring relevant information from unstructured or semistructured documents. In this process you may extract key information from a text into you internal database.

In our solution, automatic extraction was used which consisted of several steps: lexical analysis (marking, or tokenization), elimination of meaningless (stop) words, lemmatization (stemming) and weighting. For the needs of our experiments, the lexical analysis was performed using the *"Lower case filter"*. Lexical analysis is a process by which meanings are associated with specific words or other textural strings. The elimination of meaningless, non-specific words was performed using the *"Stop words filter"* in process extraction. Lemmatization was performed using the *"Stem filter"* and finally the weighting was performed using the *"Index filter"* in document extraction. All these filters are part of the *"Jbowl"* library [17]. The *"Jbowl"* library is an original software package developed in Java to support the task of processing text documents.

This library is described in more detail in [18]. The diagram of the preprocessing process of text documents is shown in Fig. 3.

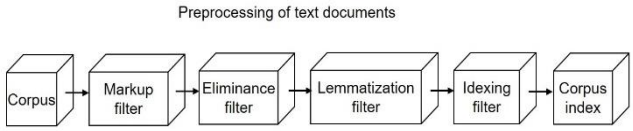

Fig. 3. Diagram of process of preprocessing text documents

Several methods are used to identify combinations of words, for example: statistical identification – words that frequently occur together in the text of the document and in the entire corpus are considered as phrases. The frequency of occurrence of the phrase (depends on the order of the individual words), the simultaneous occurrence of words (does not matter on the order) or the distance between the individual words of the phrase in the text is monitored. Since the simultaneous co-occurrence of two or more words does not necessarily mean that they form a phrase, this method is not 100% successful [18].

Another is syntactic identification – a method similar to statistical identification. The difference is that when identifying phrases between individual words of a potential phrase, their syntactic component is analyzed. Phrases are first searched for by statistical identification and their relevance is confirmed by the existence of an entry in existing phrase dictionaries or managed dictionaries.

5 Training with CoreNLP support

CoreNLP is tool for natural language processing in Java language. CoreNLP enables users to make linguistics annotation of text, named entity, parser, coreference, quote attributions, and realations. More information about modules is in thebprevious chapter.

Anonymization services use artificial intelligence technologies (CRF NER hybrid method) and NLP tools for automatic annotation and anonymization of court decisions or other unstructured documents.

The hybrid NER method relies on two data sources, first is statistical CRF model, trained on data of a compatible type of documents, second is linguistic rules and dictionaries.

Anonymization services are composed of several components in accordance with the architecture of microservices. Each component exposes its services using interfaces (REST/SOAP/GUI/Java API) and has its own responsibility. The main components include Anon-web, nlp-tools, nlp-evaluate, nlp-train, morpho-admin, more is listed at [19].

Two types of models are used in anonymization, NER model for named entity recognition and anonymization model for binary classification of named entities.

5.1 Entity data-set creator process

After performing this procedure, the so-called golden dataset that contains an ann file for each document that consists of entities that are represented by two lines that contain the entity's type, its position (start, end) and its text.

An example of the representation of entities in one file of the golden dataset is shown in Fig. 4.

```
T1      NamedEntity 41 65       Krajský súd
        v Bratislave
A1      NamedEntity_value T1 SUD
T2      NamedEntity 89 113      JUDr. XY
A2      NamedEntity_value T2 SUDCA
```

Fig. 4. A form of tagging entities of the golden dataset file

The dataset obtained in this way is subsequently divided in two ways, which are used in the verification of the quality of the model:

1. Train-test split – Division into training and test set in a ratio of 80 to 20
2. 5-fold cross validation split – Splitting the training and testing set 5 times in a ratio of 80 to 20, while each document is used for testing in turn.

5.2 Data-set training procedure

The model is trained using the nlp-train application. Nlp-train is a console application that contains the training configuration and allows training of different types of models. Several models are trained – NER model (multiple) and ANON model. The advantage of dividing the model into several parts is that the training speed increases and the memory is used more efficiently.

Command to launch training console:

```
./nlp-train-1.0.5-SNAPSHOT.jar
```

For faster training, it is recommended to set the Java parameter of the virtual machine Xmx64G via myapp.conf [20]. After starting the application, a shell console is available for entering commands, you can use the *help* command for orientation.

NER training is in several steps, as can be seen from Fig. 5. Final training is done over the entire dataset. To verify the quality of the model, 5 sets of models (for each fold in 5-fold cross validation) are trained, 4 CRF models for different groups of entities (due to memory and speed optimization).

```
train ner classpath:anon-dataset/ann-all.zip txt-train.zip tmp/ --labels-filter OSOBA_ANON, OSOBA_NORMAL, SUDCA, RODNE_CISLO, BYDLISKO, ADRESA, ANONYMIZOVANA_LOKALITA, CISLO_OP

train ner classpath:anon-dataset/ann-all.zip txt-train.zip tmp/ --labels-filter SUD,PO_S_JEDINYM_KONATELOM_FO,ORGANIZACIA,BANKA,VYROBCA_TYP_AUTA,DIC

train ner classpath:anon-dataset/ann-all.zip txt-train.zip tmp/ --labels-filter URL_ZAKON,URL_ADRESA,CISLO_PLATOBNEJ_KARTY,SPISOVA_ZNACKA

train ner classpath:anon-dataset/ann-all.zip txt-train.zip tmp/ --labels-filter LIST_VLASTNICTVA,KATASTRALNE_UZEMIE,CISLO_PARCELY,LEGISLATIVA
```

Fig. 5. Commands for training dataset subsets in CoreNLP

Ann-all.zip contains annotation information, i.e., for each decision, it contains a list of entities with the number of the beginning and end of the entity and an indication of which entity it is. The original set was created by manual annotation by several annotators. The current set was created by dumping from the ANON database and transforming it into ann format for selected decisions. *Txt-train.zip* are the texts of decisions for training, *tmp/* is the folder where the tsv files are temporarily stored, based on which the AI is finally trained.

Evaluation of automatic anonymization (ANON) performed by the procedure on a sample of 5110 chains – tokens, we obtained the mentioned data. We record very high percentages of all anonymized token metrics for annotator 3 and annotator 2. Similarly, hybrid anonymization achieved very good percentages of anonymized token metrics, except for the coverage metric. The on-court annotator achieved good percentages only in the accuracy metric and poor percentages in the F1 score and coverage metrics. To justify the obtained data in % in Tab. 1, we present the following facts:

- In the case of anonymization, it is far from clear what should be annymized and what should not be anonymized.
- The human annotator at the court missed almost of the entities during anonymization.

6 Conclusion

Anonymized documents using the automatic anonymization method, which is available on IP BAI (within IS BAI), significantly increase the quality of personal data protection in accordance with applicable legislative regulations, primarily the Act on Personal Data Protection and Amendments to Certain Acts No. 18/2018 Coll.

By preprocessing text documents and then defining training sets based on the given examples to identify the entities Person, Address (residence), Date of birth, Organization and Legislation, we created a basic set of data for machine learning. The results of the anonymized documents on these training sets are the reference data.

An important problem of ensuring anonymized data is in sufficient utility to train high-performance models. Elimination of this disadvantage is possible by better preprocessing of the anonymized text. The process of anonymization can remove valuable information which results in degradation of model accuracy and effectiveness. This is an important restriction.

We intend to improve the method of anonymizing items subject to personal data protection in several ways. One is the expansion of training sets by obtaining information from the anonymization of entities.

There are differences in the results obtained from the extracted information of the source data in the NER method. Such consistency of results ensures the resolution of duplicate data or of conflicting records in the anonymized data. Here, we used preprocessing of text documents and information extraction. More detailed information is in Chapter 4. The second is the combination of the manual anonymization method and machine learning in the processing text documents for recognizing named entities (Person, Address (residence), Social Security number, Organization, Legislation) described in Chapter 3. Currently the procedure of expanding training sets are being implemented, thereby increasing the level of data protection in anonymization services.

Acknowlegments

This article was supported by the ITMS project 21110120026, implemented by company DITEC, a. s., the project was implemented within the OPIS-2011/1.1/36-NP calls, Operational program Informatization of the company and is maintained in operation.

References

[1] Ministry of Justice of Slovak Republic, (2013). Courts Act No. 757/2004 Coll. https://obcan.justice.sk/infosud/ zoznam/rozhodnutie

[2] Ministry of Justice of Slovak Republic, (2015). Decree on the publication of court decision No. 482/2011 Coll. https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2011 /482/20120101

[3] Ministry of Justice of Slovak Republic, (2010). Court on Free Access to Information No. 211/2000 Coll. https://www.slov-lex.sk/pravne-predpisy/SK/ZZ/2000 /211/20160701

[4] P. Kvasnica, "Creation of datasets for machine learning in the anonymsation of unstructured documents". In: *Proceedings of the 22and ISC'2024 Industrial Simulation Conference*, pp. 13–18, ISBN 978-9-492589-30-3, 2024.

[5] E. M. Nrl, et al., "MUC-7 EVALUATION OF IE TECHNOLOGY", Overview of Results MUC-7 Program Committee. In *Program*, 1998.

[6] H. H. Hock, B. D. Joseph, *Language History, Language Change, and Language Relationship: An Introduction to Historical and Comparative Linguistics* [online]. [s.l.]: De Gruyter, ISBN 9783110214307, 2009.

[7] D. Hládek, et al. "The Slovak morphological classifier". *ELMAR, 2012 Proceedings*, pp. 12–14, September, 2012. Available online: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6338504\nhttp://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6338504>.

[8] O. Kaššák, *Extrakcia pomenovaných entít pre slovenský jazyk.* Znalosti 2012. Sborník příspěvků 11. ročníku konference. Praha: Matfyzpress, pp. 52-61, 2012.

[9] S. Chakrabarti, *Mining the web, Discovering knowledge form hypertext data*. Morgan Kaufmann publishers, ISBN-13: 9781558607545, 2002.

[10] K. Han, Y. Song, H. Rim, *Probabilistic Model for Definitional Question Answering*. Korea University, Seoul, Korea, 2005.

[11] B. Salton, G. Salton & C. Buckley, "Term weighting approaches in automatic text retrieval". Information Processing and Management, 24(5), pp. 513-523, 1988.

[12] S. Dumais, *Enhancing performance in LSI retrieval*. Technical Report 91/09/17, Bellcore, 1991.

[13] T. M. Mitchell, *Machine Learning*, The McGraw-Hill Companies, Inc., New York, USA, 414 ps, 1997.

[14] I. H. Witten, and E. Frank, *DATA MINING, Practical Machine Learning Tools and Techniques*, Morgan Kaufmann Publishers, 2005.

[15] P. Stenetorp, et al., "A web-based tool for NLP-assisted text annotation. *Proc. of the Demonstrations at the 13th Conf. of the European Chapter of the Association for Computational Linguistics.* ACL, pp. 102-107, 2012.

[16] A. Bagga, and B. Baldwin, "Algorithms for scoring coreference chains". *The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*, pp. 563–566, 1998.

[17] C. D. Manning, et al., "The Stanford CoreNLP Natural Language Processing Toolkit". *Proc. of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, ACL, pp. 55–60, 2014. Available online: http://aclweb.org/anthology/P14-5010.

[18] P. Bednár, P. Butka, J. Paralic, *Java Library for Support of Text Mining and Retrieval*. ZNALOSTI 2005, Stará Lesná, Vyd. Univerzity Palackého Olomouc, pp. 162-169, ISBN 80-248-0755-6, 2005.

[19] Ministry of Justice of Slovak Republic, (2010). Information on the componenets of anonymization services, accesible from the justice.sk domain. http://intranet/intranet/sudy

[20] Spring by VMWare Tanzu (2021). How to bild effective agents, Retrievel Augmented Generation. https://docs.spring.io/spring-ai/reference/api/retrieval-augmented-generation.html

Peter Kvasnica works in the Department of Information and Project Control as a project manager. He graduated from the University of Technology in Brno (VUT Brno) in digital computers. He received the degree of Doctor of Philosophy (Ph.D.) in computer science application at M. R. Štefánik Military Academy of Aviation in Košice. He received master degree (Mgr.) in public administration, implementation and control systems. He has been involved in the development of special adaptive mathematical models of objects from the point of view of programming and use of a distributed computer system in flight simulators for real-time applications. He is interested in software tools for parallel programming MPI, Open MP and OpenCL for creation of GPU applications and application of AI such as machine learning applications and text anonymization.

Received 4 April 2025