Final report on TRIADO enrichment phase

TRIADO project team, May 2019

Explanatory notes

This document presents the results of various experiments conducted in the 'enrichment phase' of the Tribunal archives as a digital research facility (TRIADO) project (February 2018-December 2018). The following question was central to this phase:

Which digital methods are most suited (in terms of quality, efficiency, etc.) for making large corpora of unstructured, imperfect data, based on analogue archive collections, suitable as a research facility? The emphasis is on enabling the application of tools from the digital domain that are already available (from laboratory to reality check) and improving access to information based on keywords relating to "who, what, where and when".

The enrichment was based on 13.8 metres of digitised material from the Central Archive of Special Jurisdiction [*Centraal Archief Bijzondere Rechtspleging*] (CABR) (167,197 scans). For this part of the project, the TRIADO project team consisted of:

- Lars Buitinck (Huygens ING/KNAW Humanities Cluster), computer engineer.
- Anne Gorter (National Archive), collection specialist.
- Edwin Klijn (Netwerk Oorlogsbronnen [*War Sources Network*]), project manager.
- Rutger van Koert (Huygens ING/KNAW Humanities Cluster), computer engineer.
- Marielle Scherer (Huygens ING), data manager.

Key findings

- With the use of Abbyy software, the quality of the OCR is sufficient to make the material searchable – with a certain margin of error. It is mainly easily legible, typed documents, such as police reports and judgments, that are well recognised by the software (approx. 85% of the words are correctly converted). These are also the most information-rich documents in the archive, with a lot of information about people, places and events.
- Abbyy and Tesseract both have their strengths. To optimise findability, a good strategy is to combine several 'layers' with OCR text.
- Preprocessing the images, machine-learning based on ground truth, and post-correction can optimise the OCR score still further.
- Automatic classification has potential. When the computer is trained using examples, types of documents with an acceptable margin of error (80% correct) are recognised.
- Named entity recognition appears to be very difficult. 'Bottom-up' extraction of individuals, locations or organisations from the CABR produces mediocre results using the currently available software. In contrast, matching existing databases with individuals, locations, organisations, etc. in the OCR of the CABR seems to work well.
- Date extraction makes sense. Further sharpening some scripts makes it possible to extract dates well (provided the quality of the OCR is sufficient).

- Potentially interesting: automatic clustering, topic modelling and SIFT matching (similarity searching).

Structure of the report

Section 1 describes the sample that was used as the basis for all the tests. Sections 2 to 5 describe the findings from the experiments on automatic text recognition, recognising 'named entities', linking to external databases, and the automatic classification of documents. Section 6 looks at the technologies that are potentially interesting, but that have only been tested to a limited extent. The report concludes with a glossary that explains the most important technical terms. All the terms printed in ***bold and italics*** in the text can be found in the glossary.

This report is based on underlying data, which are contained in separate appendices to the report:

- Appendix A Scores of test set A
- Appendix B Scores of test set B
- Appendix C Comprehensive description of the sample
- Appendix D Interface TRIADO demonstrator

1. Description of the sample

For the project, researchers worked with a sample consisting of 13.8 metres of digitised material (167,197 scans) from the Central Archive of Special Jurisdiction [*Centraal Archief Bijzondere Rechtspleging*] (CABR), which is stored at the National Archive (access number: 2.09.09). The CABR is composed of various smaller archives, for example of courts, public prosecutors and police departments. Due to the size of the whole CABR (3.8 kilometres) and the huge diversity of the archive material, it was practically impossible to make a representative selection. Moreover, it was necessary to have consistency of content with a view to the proof of concept in the second part of the project. In this phase, the research is focused on gaining a picture of the criminal careers of individuals, with particular attention to sequence and geography. The collection specialist from the National Archive and the researcher from the NIOD Institute for War, Holocaust and Genocide Studies (NIOD) made the selection jointly. The sample consists of two parts:

Part 1:
A sample was taken from 13.8 metres of digitised CABR files, which provides a good representation of the geographical distribution of the files across the Netherlands. The basis of the sample was personal files created by police departments. The CABR contains 100 series of personal files compiled by 80 police departments. One file was selected from each of these series by choosing a random inventory number from the inventory of that series. The first file under this inventory number was selected.[1] It was then established who the suspect was in the file. A search was made of the CABR database to check whether there were any more files about that person. If this was the case, the files were sought and added to the sample. In total, this sampling method produced 217 files and 1.2 metres of archive material.

Part 2:
The above sampling method turned out to be extremely time-consuming. To save time, while retaining the geographical distribution as the basic principle, a new approach was worked out. Before the transfer of the CABR to what was then the Kingdom's General Archives [*Algemeen Rijksarchief*], some processing work was carried out on the archive. As a result, some of the files[2] have their own inventory number, including the files of the judicial bodies. The identities of the individuals to whom the files refer were included in a database. From the database, a printout was made of all the judicial bodies, with the associated files and inventory numbers. The number of inventory numbers was then established for each body. It was then calculated how many inventory numbers had to be selected pro rata for each body. In this second part, a total of 1204 inventory numbers were randomly selected, consisting of around 12.6 metres of archive space.

[1] There are several files under most inventory numbers.
[2] Files compiled by the judicial bodies (the Tribunals, the Special Courts of Justice, the Special Criminal Divisions, and the Special Court of Cassation), which were established around the Netherlands.

2. OCR score

Methodology

OCR was performed on 167,122 of the 167,197 scans.[3] 75 tiff files were corrupt. Two ***OCR*** programs were used:

- Abbyy CLI OCR 11
- Tesseract 4.0.9 rc4

To measure ***OCR*** quality, two samples were put together, for which ***ground truth*** files were created:

- Set A consists of a random sample of 100 documents from the 13.8 metres of digitised material. Set A comprises various types of documents, including hand-written letters, photographs, forms and other documents that are not typewritten. In the ***ground truth***, 96 documents were transcribed.[4]
- Set B consists of a non-random sample of 150 documents from the 13.8 metres of digitised material. The documents selected are scans of police reports and judgments, that is to say exclusively typewritten material. Handwritten marginalia were not transcribed in the ***ground truth***.

It was decided to split the data into two data sets because one single random sample from the entire sample would give a distorted picture. The documents that can be considered the most informative and valuable for research – police reports and judgments – are usually typed.

Scores

Set A

	Average *WER*[5]	**Median *WER***	**Average *CER***	**Median *CER***	**Weighted WER[6]**
Abbyy	71.24	35.22	50.97	24.95	37.17
Tesseract	85.54	44.66	60.17	50.57	53.98
Tesseract	272.95	69.54	132.11	48.55	52.23

[3] Bold type indicates that the term is explained in more detail in Section 7. Glossary.
[4] The four remaining documents were photographs and other items that cannot be transcribed.
[5] All WER measurements in this report are WER-independent ('bag of words').
[6] Percentage of the total number of words of the sample that is incorrectly converted.

darkened					

Set B

	Average WER	Median WER	Average CER	Median CER	Weighted WER
Abbyy	21.64	12.33	9.86	4.62	15.62
Tesseract	89.23	39.26	60.96	29.19	55.56

There are a number of things that stand out in these scores:

- Excessively high ***WER*** and ***CER*** scores and large differences between documents, especially for Tesseract. In general, documents with a lot of typed text produce the best scores. ***Hybrid*** or handwritten documents pull the average ***CER*** and ***WER*** down sharply. This occurs mainly because Tesseract – and to a lesser extent Abbyy – tries to recognise words, especially in the case of faint ink on carbon paper. Tesseract also frequently divides words into several parts. This causes substantial differences between the number of words in the ***ground truth*** and the number of words that the software recognises. In the Tesseract scores, this sometimes leads to a WER or CER that is far above 100. Tesseract's tendency to make documents with different sorts of print (a combination of typewriter, handwriting and pre-printed text) nevertheless machine-readable increases the number of errors, but also increases the number of correctly converted words.
- Big differences between test set A and test set B. This has to do with the composition of the test sets. Set A is a random sample that includes handwritten texts, ***hybrid*** forms and other material that is difficult for the software to process. Test set B is mainly composed of police reports and judgments. These are almost exclusively typescripts with a high text density.

The general conclusion is that Abbyy produces the best initial results for typescripts. In the case of police reports, judgments and other typed materials with high text density (set B), there was a weighted score of 15.62 ***WER***. This percentage is comparable with the ***WER*** of 19.45 that was measured in the small sample of the Full Automatic Archival Access project.[7] Both margins of error are low enough to develop a full-text search function and apply data enrichment, but there is clearly still considerable room for improvement.

Ensemble approach

With some regularity, Abbyy failed to recognise something perfectly, while Tesseract did, and vice versa. In the demonstrator with which the 13.8 metres could be searched through a web interface, researchers

[7] Final report of Full Automatic Archival Access (FAAA) project (Amsterdam 2016), www.oorlogsbronnen.nl/sites/default/files/FINAL%20REPORT%20project%20Full%20Automatic%20Archival%20Access_1.pdf.

worked with various different OCR 'layers'. The 'noise' did increase (i.e., the precision declined), but as yet this disadvantage does not seem to outweigh the great advantage of higher ***recall***. While searching the corpus, the user has relatively few problems with the 'noise' in the ***OCR***, because it generally does not lead to names but rather to non-existent words.

It is expected that the findability of search terms can be greatly improved if several ***OCR*** and ***Handwritten Text Recognition (HTR)*** text layers are used in the background. To what extent this is desirable and comprehensible for an end user is something that requires further research.

Pre-processing images and machine learning

In TRIADO, researchers experimented with a method called ***adaptive gaussian thresholding***. Pixels that look like ink were marked and subsequently darkened in the original. A pixel value was then allocated, which is between the original pixel value and the pixel value belonging to black. The ***ground truth*** of set B, combined with the 'darkened' binary images, was subsequently used as the basis for the ***machine learning*** with Tesseract. Set B was randomly divided into two parts: one for training (80%) and one for testing (20%).

After darkening (see figure 1), the ***WER*** score of these adjusted images in Tesseract was initially 17.78. This is higher than the ***WER*** score of the same images without adjustment. This is because after darkening, Tesseract sees small stains, uneven patches and similar features as ink. The darkening creates more 'noise', but in the case of hard-to-see text, more words are ultimately recognised.

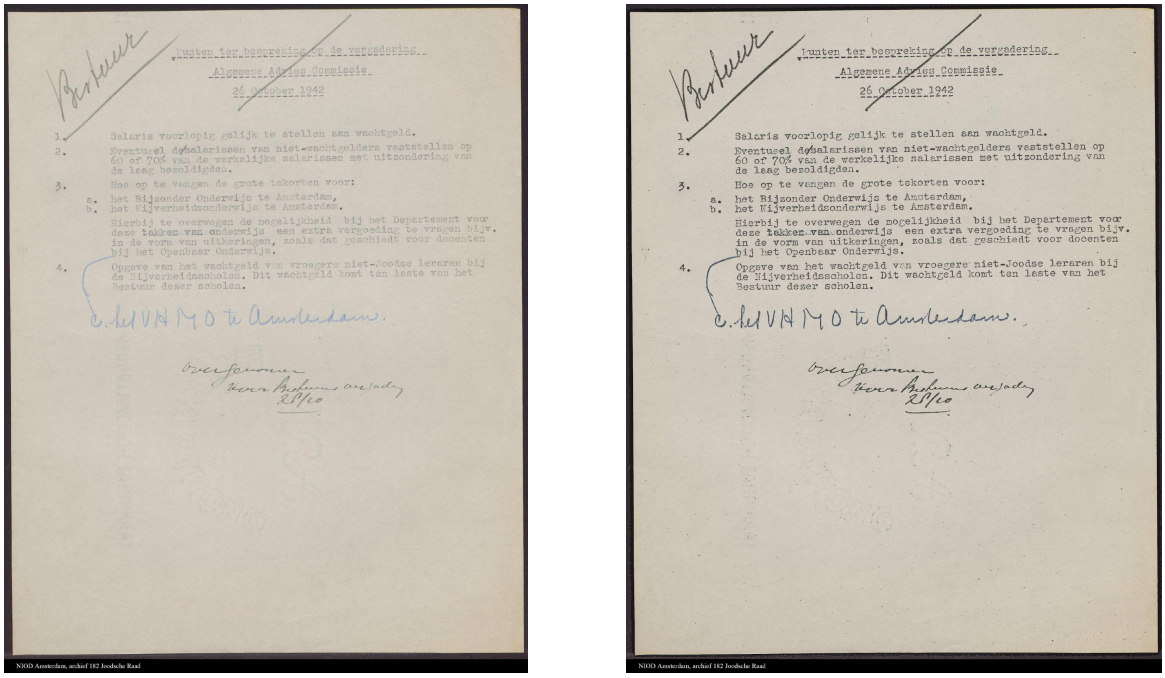

Figure 1: effect of darkening (left, the original; right, after darkening, on documents from the archive of the Jewish Council at the NIOD)

When Tesseract was trained on darkened images, the ***WER*** fell to around 10.37 (see figure 2). There was improvement up to 80,000-90,000 iterations, after which overtraining appeared to set in. Therefore, darkening in combination with machine learning appears to be a good method for countering the limitations of Tesseract.

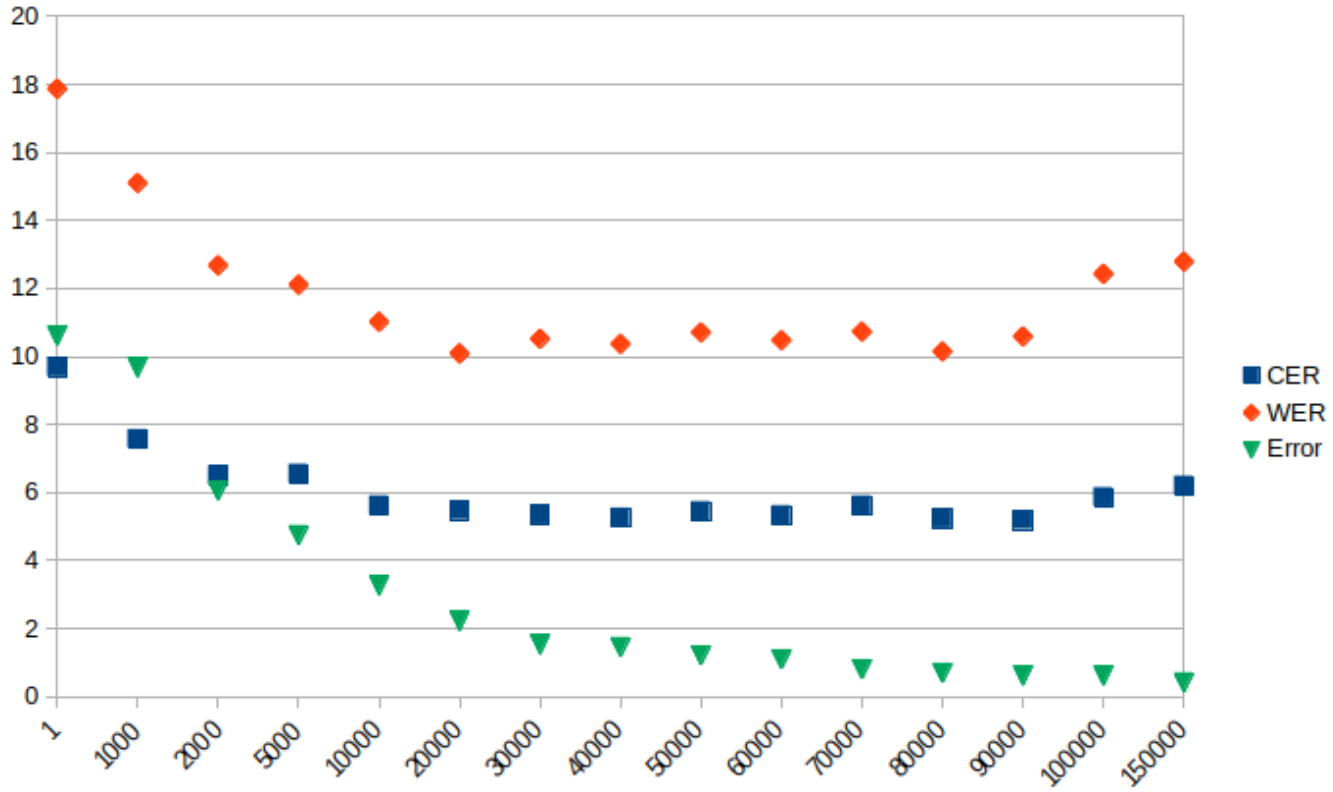

Figure 2: number of iterations in the training of Tesseract with the darkening method

Recommendations

Testing OCR results in HTR *convolutional neural network* Transkribus

This pilot involved only experiments with Abbyy and Tesseract in a secured offline environment. Due to these circumstances tests with cloud-based software were impossible. The tests using Tesseract showed that the use of ***machine learning*** to obtain better ***OCR*** can produce good results. In the READ project, ***WER*** percentages were obtained with Transkribus for handwritten sources which were lower than percentages measured in TRIADO for typed material.[8] Recently, with transcribing 17th century notarial archival documents with some manual training a CER was measured of 5.[9] Both Transkribus and

[8] https://transkribus.eu.

[9] https://www.volkskrant.nl/wetenschap/nieuw-gereedschap-voor-historici-computers-die-handgeschreven-historische-papieren-omzetten-in-digitale-tekst.

Tesseract are helped by strong training models. It is therefore advised that a test be done with Transkribus and the results compared with those from the TRIADO project.

Ensemble approach: more noise but better search results

To optimise searching, an ensemble approach – in which the results of various ***OCR*** outputs are combined – appears to be a good way of increasing findability. Admittedly, the 'noise' increases, but more words can be found.

Post-correction with piccl and ticcl

It is possible to correct frequently occurring ***OCR*** errors using automatic post-correction software (piccl and ticcl). Due to the technical circumstances of the standalone working environment of TRIADO, it was not possible to apply this software, but it usually leads to a quality improvement that is simple to achieve. Frequently occurring errors involving the letters 'l' and 'i', and 'e' and 'o', as well as words such as 'vrouw' ('woman') and the ***OCR*** equivalent 'vrouvv', can be corrected with software. This method of post-correction can be applied with the help of a ***confusion matrix***, which is a matrix that indicates the probability of a specific character being confused with another.

Adapting to the language in the original documents

The ***OCR*** software is set to recognise only a few specific languages (English, Dutch and German). However, in a document that we know is in Dutch, it makes no sense, and is even counterproductive, to add German as an option. Stains and small areas of damage in the documents can lead to a ß, for example, appearing in the transcript. If Dutch alone is put in as the document language, this kind of error can be reduced. The language that is used most in a document can be determined fairly reliably through the use of ***n-grams.*** These n-grams determine the probability that a word belongs to a particular language.

3. Automatic classification

Methodology

The CABR is a diverse archive with many different kinds of documents: forms, membership cards, typed correspondence, judgments, etc. Experiments have been conducted with ***machine learning*** to teach the computer to recognise specific types of documents automatically (***automatic classification***). The results of this work were converted into filters ('document type') in the demonstrator.

From the 13.8-metre sample, a non-random selection of 4768 scans was divided manually into types of documents. From this ***ground truth*** set, 28 classes (see figure 4) were further developed by collecting 20 or more examples and training the computer to recognise those documents. These were classes with frequently occurring documents (e.g. extracts from police reports, verdicts and witness statements). Of these data, 80% were used for training and the remaining 20% for testing. For identifying types of documents, the '***random forest method'***, as it is known, was used on the textual content (Abbyy OCR) of the files.[10] In addition, ***deep learning*** was also carried out on the layout of the documents. The scans

[10] P. Geurts, D. Ernst and L. Wehenkel, 2006. Extremely randomized trees. In *Machine Learning* 63(1): 3-42.

were binarised (made black and white) and reduced to a maximum of 150x150 pixels while retaining the aspect ratio. These adjusted images were subsequently imported into a ***convolutional neural network*** that was modelled on the work of Le Kang and made use of DL4J.[11]

Scores

Initially, the overall ***accuracy rate*** of the automatic classification was a little under 70%. Following ***machine learning,*** it was possible to improve this to 80% (see figure 3). It therefore appears to make sense to use ***machine-learning*** to recognise classes. The learning curve was still rising at the end of the experiment, which means that with more training data, the score could have been a little higher.

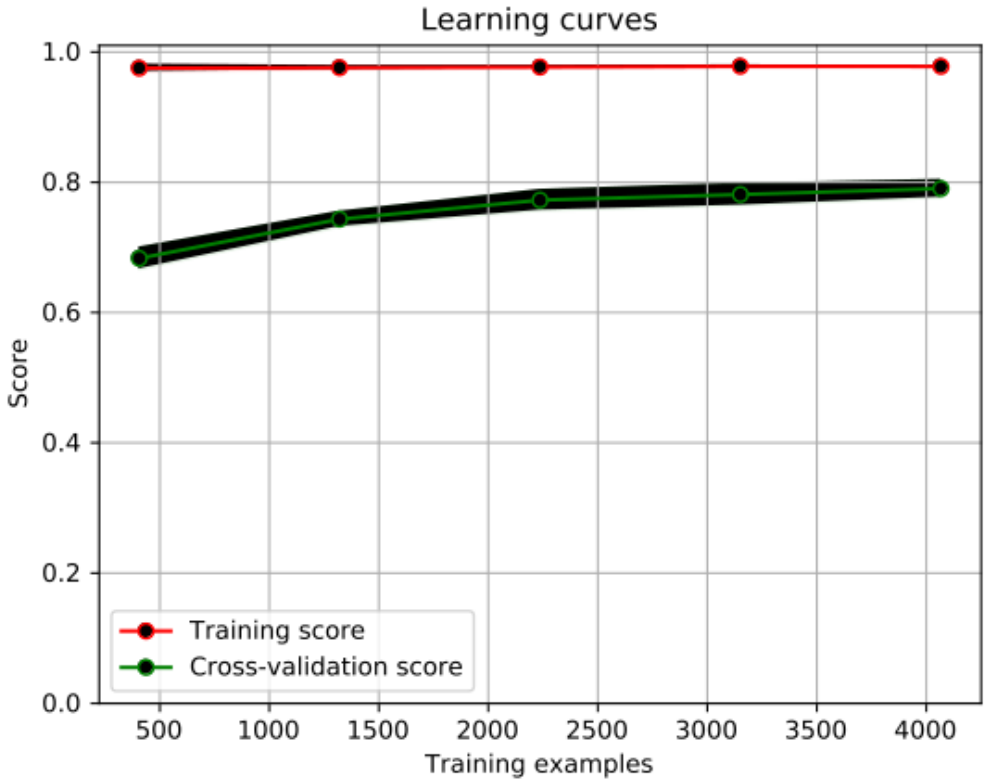

Figure 3: effect of machine learning on accuracy rate of automatic classification (set B)

The ***confusion matrix*** (see figure 4), as it is called, shows the differences by 'class' between the computer interpretation (predicated class) and the ***ground truth*** (actual class). Sometimes an incorrect interpretation is easy to explain. For example, in the police report class (class 13), 158 of the 211 pages were recognised. The computer recognised the page as the type 'correspondence_typed' (class 3) in 23 cases. Because these documents do indeed look very similar, this is a deviation that is easy to explain.

[11] Le Kang, Jayant Kumar, Peng Ye, Yi Li, David S. Doermann, Convolutional Neural Networks for Document Image Classification, 2014. In *ICPR '14 Proceedings of the 2014 22nd International Conference on Pattern Recognition*. See also https://deeplearning4j.org.

		Predicted Class																												
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	Total
Actual Class	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
	1	0	0	0	3	0	1	0	0	0	0	0	0	0	3	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	9
	2	0	0	0	0	0	3	0	0	0	0	0	0	1	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9
	3	0	0	0	75	0	7	4	0	0	0	0	0	1	33	1	1	0	0	0	3	3	0	0	0	2	0	0	0	1	131
	4	0	0	0	8	1	2	0	0	0	0	0	0	0	11	0	2	0	0	0	0	1	0	0	0	0	0	0	0	0	25
	5	0	0	0	2	1	22	2	0	0	0	0	0	1	12	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	42
	6	0	0	0	2	0	5	17	2	0	0	0	0	0	12	0	4	0	0	0	1	0	0	0	0	0	0	0	0	0	43
	7	0	0	0	9	0	2	2	2	0	0	0	0	0	4	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	21
	8	0	0	0	3	0	2	0	0	0	0	0	0	0	2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	8
	9	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
	10	0	0	0	0	0	0	0	0	0	0	30	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	31
	11	0	0	0	0	0	1	0	0	0	0	0	0	0	2	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	5
	12	0	0	0	2	0	6	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10
	13	0	0	0	23	0	7	1	1	0	0	2	0	0	158	3	10	0	0	0	0	1	0	2	0	2	0	1	0	0	211
	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11
	15	0	0	0	3	0	1	0	0	0	0	0	0	0	11	0	26	0	0	0	0	1	0	0	0	0	0	0	0	0	42
	16	0	0	0	2	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4
	17	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	8
	18	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
	19	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	4
	20	0	0	0	2	0	0	1	2	0	0	0	0	0	2	1	0	0	0	0	0	7	0	0	0	0	0	0	0	1	16
	21	0	0	0	4	0	1	3	0	0	0	0	1	0	23	1	2	0	0	0	1	0	1	1	0	0	0	0	0	0	38
	22	0	0	0	10	0	0	0	0	0	0	0	0	0	2	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	15
	23	0	0	0	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	4
	24	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	7
	25	0	0	0	9	0	4	0	0	0	0	0	0	0	5	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	21
	26	0	0	0	2	1	0	0	0	0	0	0	0	0	4	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	9
	27	0	0	0	2	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	5
	28	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2
	Total	0	0	0	165	3	66	31	9	0	3	32	1	3	307	21	54	2	0	1	11	13	2	4	0	7	0	1	0	3	

Class no.	Description	Class no.	Description
0	beslissing_voorwaardelijke buitenvervolgstelling	15	rapport
1	besluit	16	rapport_pra
2	bevel_tot_dagvaarding	17	sententie
3	correspondentie_getypt	18	soldbuch
4	correspondentie_handgeschreven	19	staat_van_dienst_in_beweging
5	dagvaarding	20	staat_van_inlichtingen
6	gerechtelijk_schrijven	21	uitspraak
7	handgeschreven_tekst	22	uittreksel_burgerlijke_stand
8	inventaris	23	verhoor_beschuldigde
9	maandcontributie	24	verhoor_van_getuigen
10	manuscript	25	verklaring_getypt
11	openbare_terechtzitting	26	verklaring_handgeschreven

12	oproeping	27	verzoek_inlichtingen
13	proces_verbaal	28	vragenlijst
14	proces_verbaal_uittreksel		

Figure 4: confusion matrix with scores of automatic classification (set B)

Recommendations

More training, ensemble learning and input from users

Automatic classification algorithms can be further improved through a smarter combination of textual and visual characteristics (***ensemble learning***). The ***accuracy rate*** of approx. 80% seems to be a promising result, considering the fact that training curve was still rising at the end of the experiment. More training is expected to further improve the ***accuracy rate***. However, raining the computer for certain classes takes a lot of machine time. It is estimated that the CABR consists of more than 100 document types, so that in a follow-up project, a selection should be made based on the importance of the content.

Reconstruction of the court proceedings

If 'classes' could be recognised with an acceptable margin of error, it would also be possible to filter the specific documents out of the files (summons, police reports, judgment, appeal) for an end user.

4. Named entity recognition

Method

FROG-NER software with standard settings was used to recognise named entities (people, organisations, locations, products, events, and other) in the ***ground truth*** files of set A and set B. [12] Due to the technical environment limitations, the ***tokenizer*** was replaced by that of Apache OpenNLP.[13]

For the purpose of measuring quality, all ***named entities*** were manually tagged using the software tool BRAT, in accordance with the instruction manual of Hendricks et al.[14] Subsequently, the following three variables were measured for set A and set B:

- The '***precision***' values (true positives/(true positives+false positives). This indicator answers the question: how much of what we find is correct?
- The '***recall***' value (true positives)/(true positives+false negatives). This indicator answers the question: how many of all the named entities from the 'ground truth' do we retrieve?

[12] https://languagemachines.github.io/frog.
[13] https://opennlp.apache.org.
[14] For BRAT see http://brat.nlplab.org.Iris Hendrickx, Antal van den Bosch, Maarten van Gompel, Ko van der Sloot and Walter Daelemans, Frog. A Natural Language Processing Suite for Dutch. Centre for Language Studies and Centre for Speech and Language Technology, Radboud University, *CLST Technical Report* 16-02, https://github.com/LanguageMachines/frog/raw/master/docs/frogmanual.pdf.

- The ***F1 value*** (2*precision*recall)/(precision+recall). This indicator gives a harmonic mean between the two.

All values range from 0 (all false) to 1 (perfect)

Scores

	Set A	**Set B**
Precision	0.271	0.281
Recall	0.253	0.387
F1	0.261	0.325

Figure 5: results of named entity recognition with FROG-NER

The ***NER*** scores (see figure 5) were low for both set A and set B. Set B scored slightly better, relatively speaking, because it consists of running text and the ***NER*** software used is trained on newspaper articles. There are various reasons for the low score. Modern newspaper articles, which FROG-NER works with particularly well, deviate in their use of language from the World War II period. For example, the use of capital letters is different: names of job titles such as 'Wachtmeester' (Sergeant) were often written with a capital letter, whereas in the Netherlands today, it is no longer written that way. Other errors were caused by German nouns, which were capitalised.

Enrichment with existing lists

Matching existing name files seems to work much better to date. In this pilot, experiments were done using the databases of the National Database of Victims of Persecution [*Nationale Database Vervolgings Slachtoffers*] (NDVS), the War Graves Foundation [*Oorlogsgravenstichting*] (OGS), and the CABR (database of suspected persons). Researchers experimented with the matching of name lists on three different text layers (based on Abbyy, Tesseract and 'Tesseract with darkening and extra training').Despite the limited test period, it was shown to be possible to identify positively a few victims in the 13.8-metre sample. Researchers also succeeded in identifying suspects who were mentioned in several CABR files.

The experiments were mainly exploratory. For example, no precise measurements were carried out. Furthermore, researchers worked with fairly rudimentary matching strategies that have not yet been refined so as to anticipate varying forms of names, minor spelling mistakes and/or ***OCR*** errors, place and date of birth in the vicinity of the name, etc. A matcher was made which tried to make a link between the names in the raw ***OCR*** and those in the databases, based on ***Levenshtein distance*** and a ***confusion matrix***. Whereas in standard ***Levenshtein*** functions, the distance between 'i' and '1' will be one, the TRIADO matcher uses a ***confusion matrix*** in which replacing one letter with a specific other letter is quantified in the degree of probability. In this way, frequently occurring ***OCR*** errors can largely be

avoided by assigning them low 'replacement costs'. For example, we define replacing '1' with 'i' and vice versa on 0.1. After this, matches are searched for with only very low adjusted ***Levenshtein distances***. This method can also be used for 'sounds like' variants.

Some lists contain all the official first names, which makes matching difficult. A text can name an 'Albert Klein' who is called 'Albertus Johannes Klein' in a list. It is possible to match these by using normalisation techniques, particularly if, for example, the date of birth or place of birth – often given in police reports – are included in the calculation of probability.

Recommendations

More training of NER software

NER is mainly intended for the 'bottom-up' extraction of names from databases. Given the nature of the CABR archive, we can expect that it will continue to be complex to retrieve named entities from the databases, but training FROG for historical documents from the period 1930-1950 would be of benefit to the results.

Disambiguating named entities

An addition to the NER is to make links with existing databases/lists of people, organisations, geographical locations, etc. With the help of ***regular expressions***, named entities in 'predictable' sentence constructions can automatically be extracted and linked to other files. Tracing named entities to one person or one location (disambiguation) is difficult, but desirable. For example, by making regular expressions that can extract name/date of birth/place of birth (usually a unique combination) from police reports, matching with external files can be improved still further.

5. Date extraction

Method

The software tool BRAT was configured to mark dates. Next, the following variants were tagged in the ***ground truth*** documents of set A and set B: Day/month/year; Month/year; Day/month; Year.

Afterwards, dates were searched for with a simple script in the ***ground truth*** of set A and set B. The result of this was compared with the manually tagged dates.

Scores

	Set A	Set B
Precision	0.688	0.571
Recall	0.284	0.707
F1	0.402	0.632

Figure 6: results of date extraction

The results of the date extraction are generally encouraging. The high degree of predictability resulted in dates – in their various forms – being fairly easy to extract from the ground truth of both sets using the software. The figures of set A and set B are close together; only the ***recall*** deviates for unknown reasons.

Recommendations

Further enhancement of extraction script

Machine learning can be applied to improve the software. Post-correction of ***OCR***, whether or not in combination with a ***confusion matrix***, can reduce the margin of error, as the predictability of dates is very high.

6. Experimental

Topic modelling

Method

Topic modelling software uses statistical models to find 'topics' or 'subjects' in collections containing documents. It is a handy way to obtain a sort of brief summary in keywords of the content of one or more documents. Two topic modelling methods were applied on the basis of the OCR'ed documents in test set B. ***Non-negative matrix factorisation*** (NMF, Lee and Seung 1999) and ***Latent Dirichlet Allocation*** (LDA, Blei et al. 2003).[15] Both are based on a clustering of documents into topics – subjects which are themselves distributed across words. Each document belongs to a greater or lesser extent to each of the topics, and this is equally true of every word that appears in the document collection. For practical reasons, each page was considered to be a document for the test in TRIADO. Because the files contained a lot of 'noise', not only filler words were removed beforehand, but also all words of two characters or fewer. ***NMF*** and ***LDA*** were used on the resulting documents.

[15] D. D. Lee and H. S. Seung (1999). Learning the parts of objects by non-negative matrix factorization. *Nature* 401: 788–791 and D. M. Blei, A. Y. Ng and M. I. Jordan (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3: 993-1022.

Results

The results obtained with ***NMF*** show interesting patterns. The 30 words with the highest values are shown for each topic. For reasons of privacy, personal names were removed from the topics. A total of 19 topics were ultimately identified. A number of these were shown to give a good impression of the content:

Topic #6: beweging lid geboortedatum godsdienst geboorteplaats onderwijs adres auto functie dienst naam genoten voornamen datum rang diploma voluit arbeidsdienst motorrijder stamboek contributie lidmaatschap wapen partij opleiding zakboekje werkzaam eereteeken front ster[16]

Topic #16: zyn volken cultuur invloed volk athene stryd zoo caesar grieken rome groote indo geest land oude tyd nieuwe ryk staat geheel griekenland eeuw ran leger macht hyk tusschen verval blykt[17]

Topic #17: groningen landwachters <naam> <naam> gearresteerd slochteren <naam> siddeburen ondergedoken woning landwacht <naam> arrestatie <naam> overgebracht onderduikers west gemeente <naam> schildwolde duitsland onderduiker huiszoeking getuige landbouwer personen boerderij wonende <naam> <naam>[18]

OCR errors cause a certain amount of noise in the topics, but not so much that they become unusable. On the contrary, the multiple occurrence of words ensures that erroneous ***OCR*** interpretations are automatically excluded.

The results for the ***LDA*** model were somewhat disappointing, possibly due to the large number of parameters that can be set. For this experiment, there was a lack of time to adjust the software precisely.

Recommendations

Limit experiments to documents with high text density and test LDA further

Topic modelling is useful for typed documents with high text density. This method could be improved if it were combined with ***automatic classification*** of documents such as judgments or police reports. ***NMF*** produced the most usable results in the test, but ***LDA*** should be tested more.

Automatic clustering

Automatic clustering is a method that makes it possible to classify similar documents based on textual and visual characteristics, without ***ground truth*** or variables being determined beforehand. Because experiments using this method require considerable computing power and there was limited capacity

[16] NSB (Dutch National Socialist Movement) membership booklet.

[17] Typescript of a secondary school textbook on national history compiled by a convinced National Socialist.

[18] Incident connected with the betrayal of people in hiding in the Province of Groningen.

available for the TRIADO project, it was not possible to carry out tests with it. However, the CABR potentially appears to be extremely suitable for automatic clustering with the help of, for example, a ***DIABOLO auto encoder*** or, more precisely, ***similarity search/scale invariant feature transform matching (SIFT matching)***.[19] With ***SIFT matching*** you can, for example, have end users search by similar documents or similar parts of documents (letterhead, family coat of arms, etc.).

7. Glossary

Adaptive Gaussian thresholding: a method with which grayscale in an image is converted into binary values (black or white), so that the foreground and background can be distinguished from each other.

Automatic classification: a set of techniques for categorising documents into a number of previously determined classes, without a human being having to be involved. The software first has to be trained for this.

Automatic clustering: a method that makes it possible to group similar documents based on textual and visual characteristics, without ground truth, classes or variables being determined beforehand. For this method, the software does not have to be trained first.

CER (Character Error Rate): the quotient of the number of characters misinterpreted by the OCR and the length of the text.

Confusion matrix: a table that shows how many classifications are true (true positives, true negatives) and how many are false (false positives, false negatives), and where the system has switched classifications.

Convolutional neural network (ConvNet, CNN): a type of deep neural network that is used a great deal in analysing and classifying images. ConvNets are frequently used in recognising faces, objects and traffic signs.

DIABOLO autoencoder: a type of artificial neural network. This encoder tries to learn by reducing its input and then recreating its original input based on the reduced information.

Ground truth: information established through human observation, for example text transcribed by humans or labels assigned by humans. This is used to train software and to determine how good the result of automated analyses is.

Handwritten Text Recognition (HTR): the ability of a computer to interpret handwritten text.

Hybrid documents: documents containing text that is partly typed and partly handwritten.

Latent Dirichlet Allocation (LDA): a statistical model for topic modelling, which makes it possible, through unobserved groups of observation tests, to explain why some data are comparable. The data are modelled as Dirichlet distributions, that is to say, statistical modelling to distinguish specific groups

[19] https://www.cs.ubc.ca/~lowe/papers/iccv99.pdf.

of documents from one another on the basis of subject (topic). Here, the topic can best be seen as a number of keywords that link the various documents.

Levenshtein distance: an indicator that gives the minimum number of operations needed to change the one string of characters into the other. For example, if you search with a Levenshtein distance of 2, two characters in a word are allowed to deviate.

Named entity recognition (NER): a way of extracting named entities (e.g. individuals, places, organisations, brands, etc.) from unstructured text.

N-gram: a consecutive sequence of *n* items from a specific piece of written or spoken text.

Non-negative matrix factorization (NMF): a topic-modelling method with which the content of text documents is clustered into 'topics' – subjects, which are themselves distributed across into words.

Optical Character Recognition (OCR): the conversion of images of typed or handwritten text into machine-encoded text.

Random forest method: a method for training classification and regression with several decision trees. The random forest method ensures that the decision trees are not trained too specifically on the test dataset.

Regular expressions: a way of describing patterns, through which a computer can identify text by means of software.

Scale-invariant feature transform (SIFT) matching: a way of finding similar images through specific characteristics.

Tokenizer: software that ensures that characters are segmented into words.

Topic Modelling: a method that applies a matrix comparison 'topics', i.e. subjects, from the text of one or more documents.

Word Error Rate (WER): the quotient of the number of words misinterpreted by the OCR and the total number of words in the text.

WER-independent or 'bag of words': a simplified model in natural language processing, in which grammar and word order are ignored. Only the number of times that a word occurs is important. This model is generally used in document classification.