Page Embeddings: Extracting and Classifying Historical Documents with Generic Vector Representations

Carsten Schnober[1,*], Renate Smit[2], Manjusha Kuruppath[2], Kay Pepping[2], Leon van Wissen[3] and Lodewijk Petram[2]

[1]*Netherlands eScience Center, Amsterdam, The Netherlands*
[2]*Huygens Institute, Royal Netherlands Academy of Arts and Sciences (KNAW), Amsterdam, The Netherlands*
[3]*Faculty of Humanities, University of Amsterdam (UvA), The Netherlands*

Abstract
We propose a neural network architecture designed to generate region and page embeddings for boundary detection and classification of documents within a large and heterogeneous historical archive. Our approach is versatile and can be applied to other tasks and datasets. This method enhances the accessibility of historical archives and promotes a more inclusive utilization of historical materials.

Keywords
Natural Language Processing, Machine Learning, Sequence Tagging, Document Metadata Enhancement

1. Introduction

From its founding in 1602 until its demise at the end of the eighteenth century, the VOC[1] engaged in long-distance trade between Asia and Europe. Additionally, within Asia, it competed with local shippers and merchants, and attempted to assert its influence over a vast region surrounding the Indian Ocean, centered around modern-day Indonesia. Today, the company is renowned for its modern organizational structure and notorious for its brutal conduct, including active engagement in the slave trade. The company's bureaucracy required detailed reports of all activities in Asia. As a result, hundreds of thousands of documents (as shown in Figure 1) were drawn up in all the company's Asian outposts, copied in Batavia, bundled, and sent to the Netherlands, where they are now preserved in the National Archives.

CHR 2024: Computational Humanities Research Conference, December 4–6, 2024, Aarhus, Denmark

*Corresponding author.
c.schnober@esciencecenter.nl (C. Schnober); renate.smit@huygens.knaw.nl (R. Smit); manjusha.kuruppath@huygens.knaw.nl (M. Kuruppath); kay.pepping@huygens.knaw.nl (K. Pepping); l.vanwissen@uva.nl (L. van Wissen); lodewijk.petram@huygens.knaw.nl (L. Petram)
https://www.esciencecenter.nl/team/carsten-schnober/ (C. Schnober); https://www.huygens.knaw.nl/en/medewerkers/manjusha-kuruppath-2/ (M. Kuruppath); https://www.huygens.knaw.nl/en/medewerkers/kay-pepping-2/ (K. Pepping); https://www.uva.nl/en/profile/w/i/l.vanwissen/l.van-wissen.html (L. van Wissen); https://www.huygens.knaw.nl/medewerkers/lodewijk-petram/ (L. Petram)
0000-0001-9139-1577 (C. Schnober); 0009-0005-1070-636X (R. Smit); 0009-0002-8032-7013 (M. Kuruppath); 0000-0002-3747-706X (K. Pepping); 0000-0001-8672-025X (L. van Wissen); 0000-0002-7246-4176 (L. Petram)

[1] Dutch East India Company; Dutch: *Vereenigde Oostindische Compagnie*

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

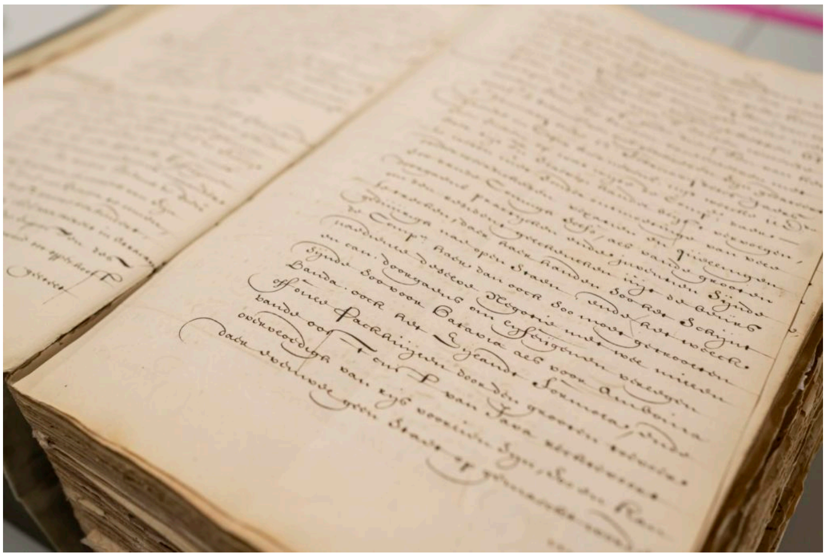


Figure 1: A page from the so-called ‘General Letters’, a collection of summarising reports within the *Overgekomen Brieven en Papieren* (English: “Letters and papers received”). *Photo: Dave Straatmeyer*

Advances in HTR (*Handwritten Text Recognition*) technology have enabled the digitization of handwritten texts that had previously only been readable by humans, often requiring a special training. Since 2019, Transkribus [9] and Loghi [12] have been used to automatically transcribe the contents of the VOC archives [22, 11].

In order to make the contents of the archive even more accessible, the task at hand is to identify the boundaries between the different documents in the archival inventories and to classify them. This poses challenges in defining what a document is, assessing the reusability of traditional finding aids, such as the one created by the TANAP project[2] (*Towards A New Age of Partnership*, 1999-2007) [1, 21], and creating a useful categorization for documents. These tasks are hugely important because they promote a more inclusive use of archival materials. Users no longer have to rely on existing indices, often created from the point of view of ruling institutions, and in the case of the VOC archives, the colonizer. Our approach helps to make certain kinds of documents, such as letters from local rulers which have never been indexed individually, more findable.

Both our source code[3] and the data [22] are publicly available.

2. Data Model and Embeddings

We present a stacked embedding model for vectorizing digitized scans of historical documents, as done e.g. for combining text and images [10] or different models [23]. We use representations of regions (region embeddings) as building blocks for vectorized page representations (page embeddings).

[2] TANAP description (in Dutch): https://www.historici.nl/resource/tanap-towards-a-new-age-of-partnership/

[3] Source code: https://github.com/LAHTeR/document_segmentation/

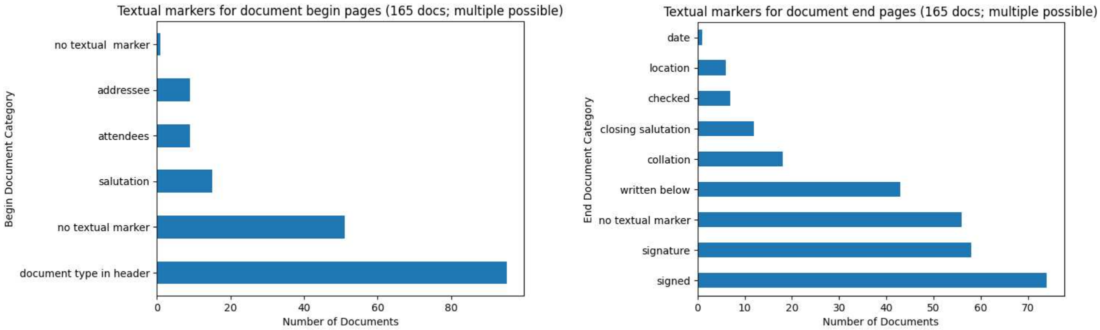


Figure 2: Textual features that indicate document begin or end pages, based on manual analyses.

Region representations are generated from the output of the Loghi HTR system [12]. They can be derived from other systems, and can be generalized to any comparable workflow.

Region Representations

Our HTR system works on the level of scans, each comprising one (default) or two pages. A scan is divided into regions, with the following information elements:

- *Text*: the text lines extracted from a region.
- *Type*: one of ten possible categories such as 'paragraph', 'page-number', 'signature-mark'.
- *Coordinates*: a list of two-dimensional coordinates that define the contour of a region.

With our primary task of document boundary detection (Section 3) in mind, we performed a manual analysis of 80 random documents to understand which features indicate document boundaries. We identified a few clear **textual** patterns that indicate document beginnings, such as salutations, lists of attendees or addressees, or the explicit mention of the document type in a page header. Document endings are often indicated by signatures, closing salutations etc. (Figure 2). In roughly a quarter of the investigated documents, no textual clues explicitly signalling document boundaries could be identified.

Visual clues were more dispersed across individual instances, e.g. the presence of page numbers on a page or large initials (Figures 5, 6). None of those clues could be derived from a region without its context. Therefore, we decided to use only the *text* and the *type* features from the list above, while skipping the *coordinates*.

The text is embedded through a language model. For the latter, we use a SentenceBERT model [17] for Dutch[4], based on *RobBERT-2022* [4]. In our initial task (Section 3), we have compared the SentenceBERT results to using GysBert [13] v2[5], a standard BERT model [5] for historic Dutch. As described in [5], we use the special `[CLS]` token to represent the text of a region. Ultimately, we concatenate the region type and the text embeddings to form a region embedding.

[4] https://huggingface.co/NetherlandsForensicInstitute/robbert-2022-dutch-sentence-transformers

[5] GysBert-v2: https://huggingface.co/emanjavacas/GysBERT-v2

The SentenceBERT model clearly outperforms the GysBertv2 approach (Table 1), while requiring significantly less memory.

Page Representations

Each region embeddings in a page is fed into a bi-directional LSTM layer [7, 20] and a linear layer, which generates a vector representation of the entire page.

The LSTM layer iterates over the region embeddings in the order specified by the HTR output. There are, however, special layout arrangements including marginalia, columns, injections etc. that make the choice of the reading order subject to interpretation and use case.

The resulting page embeddings serve as input for document boundary identification and document classification (Sections 3, 4).

3. Document Boundary Detection

Related works tasks are highly data-specific and have been tailored towards modern business documents [15, 6].

In our context, a *document* is defined as a sequence of n pages with a begin page, an end page, and ≥ 0 pages in between (*INSIDE*). Document lengths vary between a single page and 800 pages.

An inventory comprises between 155 and 2655 pages, with an average of 885. It also contains pages that are not part of a document, for instance empty pages, covers, or tables of contents. This fits the established *IOB* schema for sequence tagging (*INSIDE-OUTSIDE-BEGIN*) [16]. From our annotations, we additionally have markers for the *END* pages of each document.

This page-based conceptualization fails to model more fine-grained cases in which a document begins on the same page as another document ends, with up to three documents in our annotations. Given that there is no objectively correct order of regions (see Section 2), annotating on the region level would require a drastically increased annotation effort with multiple annotators, which makes the generation of meaningful amounts of training data practically impossible.

Training Data

As a **primary dataset**, we have manually annotated all pages of 16 randomly selected complete inventories from the VOC archives for the purpose of training a machine learning sequence tagger.

From a user's perspective, detecting the document boundaries is the most important part of the task, as they segment an inventory into usable units, i.e. documents. As indicated in Sections 1 and 2, the definition of a document is inherently ambiguous and use case-dependent. While meaningful from an archival perspective, the documents defined in the context of the TANAP project [21] turned out too coarse-grained for the purpose of historical research which focusses on content rather than chronological or administrative document boundaries. Therefore, the annotations made for this work follow a more fine-grained definition of documents.

In order to augment our data with a **secondary** and **tertiary dataset**, we have re-used two large sets of annotations that were created for unrelated purposes. Instead of annotating inventories, the annotators focussed on finding specific documents within inventories and marked their respective boundaries. Because not all documents in an inventory were annotated, we cannot make assumptions about un-annotated pages, hence there are no *OUTSIDE* pages available from these annotations. In order to approximate a realistic context, we have added blank *OUTSIDE* pages around those documents.

Furthermore, a part of these additional data (the **secondary dataset**) has originally been annotated for research about a specific category of documents (*Generale Missiven*). These documents happen to be extraordinarily long, follow specific conventions, and cover specific topics. Initial experiments have shown that adding those hundreds of non-representative documents results in low accuracy for identifying other types of documents. To prevent that skew, we have used random sub-samples of the secondary and tertiary datasets respectively equal to the size of our primary dataset. The union of these three datasets result in our total training dataset.

In total, the annotated dataset we use for training and validation comprises 12,000 pages from 48 inventories. Roughly 8,200 of them are *INSIDE* pages, 2,200 boundary pages, and 1,800 *OUTSIDE* pages.

Data Model

The boundary pages include 1,000 plain *BEGIN* and *END* pages respectively, plus 200 that are both: pages on which one or more documents end, and another one starts. In an initial experiment, we trained a classifier that explicitly models each of these as separate classes. It achieved a precision of only 0.06 on these pages. Qualitative analysis quickly revealed that these pages were hardly distinguishable from others in terms of content and context, which explains the catastrophic performance.

We adapted our data model to merge problematic categories without compromising too much on the usefulness. The result is a slight variation of the original *IOB* format that identifies documents by identifying *INSIDE*, *OUTSIDE* and *BOUNDARY* pages; with the latter unifying *BEGIN* and *END* pages. Conceptually, this workaround results in a schema that resembles *IOB*.

Machine Learning Model

We use the page embeddings introduced in Section 2 as input to another bi-directional LSTM layer [7] and the page labels introduced above as objectives for optimizing the neural network weights. The cross entropy loss [14] is weighted by inverse class frequencies to balance out the skewed distribution of page classes in the dataset.

The output of the LSTM is passed through a standard linear layer and a softmax layer [3] to determine the output class per page. Figure 3 provides a schematic illustration of the model. The final output additionally passes a set of simple heuristics to avoid impossible output sequences such as *INSIDE-OUTSIDE* and *OUTSIDE-INSIDE* – there must always be a *BOUNDARY* page to indicate a document beginning or ending. This heuristic approach is skipped during training.

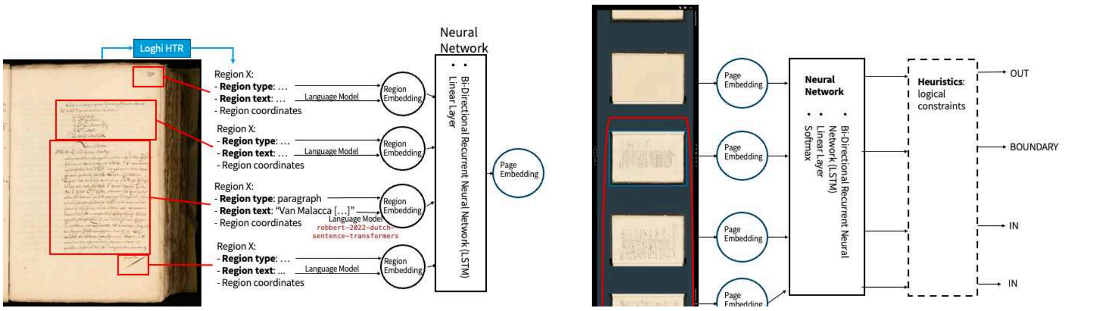


Figure 3: Region and Page Embeddings as Input for a Neural Network for Document Boundary Detection

In more complex sequence tagging tasks like Named Entity Recognition (NER), state-of-the-art models often combine an LSTM layer with an additional Conditional Random Field (CRF) [8] to model transition probabilities, rendering illogical sequences improbable. Our task, however, does not have different classes per page type, resulting in much fewer possible sequences so that we can define constraints heuristically instead of applying a CRF.

We have used region and page embedding sizes of 128 and 64 parameters respectively. Both LSTM bi-directional layers use 64 parameters as well. We iteratively increased all these configurations up to 512 parameters per layer. Those changes did not lead to changes in the results, so we use the smallest network architecture for minimizing resource consumption. All results are thus based on the 128/64 embedding and layer sizes.

The lion's share of the training time is consumed during the inference of the text embeddings. Since we do not adapt the language model weights, we can cache the text embeddings during the first training iteration which enables us running many training epochs within seconds. The model performance converged after 9 to 10 epochs – roughly 5 minutes on a consumer laptop –, so we stopped the training after 50 epochs.

Results

We have evaluated our model by randomly sampling 80% of the three datasets for training, and 20% for validation respectively. Table 1 shows the total results per page type and per dataset.

The division illustrates a clear difference in performance per sub-dataset: while detecting boundaries for the *Generale Missiven* dataset is very accurate, the other datasets contain less homogenous document types and consequently yield significantly lower results. This is confirmed by initial experiments in which we trained a model only on the *Generale Missiven* dataset, which achieved precision and recall scores close to 1.0 for all page types.

Qualitative analyses indicate, not surprisingly, that the data samples for which the model performs best use more standardized language, such as formulaic document beginnings and endings.

Table 1
Document Boundary Detection Results

Dataset	Page Type	SentenceBERT			Gysbert-v2			#Pages
		F1	Prec.	Rec.	F1	Prec.	Rec.	
Total	*INSIDE*	0.87	0.78	0.98	0.85	0.79	0.92	1,324
	OUTSIDE	0.75	0.94	0.63	0.94	0.92	0.95	403
	BOUNDARY	0.54	0.7	0.44	0.39	0.57	0.3	510
Primary	*INSIDE*	0.84	0.74	0.98	0.83	0.74	0.94	1,046
	OUTSIDE	0.74	0.96	0.6	0.95	0.95	0.94	375
	BOUNDARY	0.5	0.67	0.4	0.32	0.61	0.22	464
Secondary (*Generale Missiven*)	*INSIDE*	0.99	0.98	0.99	0.92	0.98	0.87	248
	OUTSIDE	0.93	0.87	1	0.95	0.91	1	20
	BOUNDARY	0.86	0.91	0.82	0.62	0.49	0.84	38
Tertiary	*INSIDE*	0.91	1	0.83	0.82	1	0.7	30
	OUTSIDE	0.84	0.72	1	0.84	0.73	1	8
	BOUNDARY	0.89	0.8	1	0.73	0.57	1	8

4. Document Classification

As another use case, we use document classification which is the task of assigning a label to a document, again defined a sequence of ≥ 1 pages.

The TANAP project [1] developed a categorization schema comprising 14 document main classes, each divided into 2 to 23 sub-classes, resulting in a total of 164 classes. These categories mirrored the VOC's focus on administrative aspects, e.g. distinguishing between letters sent to the Netherlands or within Asia. However, the large number of fine-grained sub-categories often led to overlapping and ambiguous categorizations, which imposes additional difficulties for both human annotators and a machine learning system. Therefore, we developed another categorization system with 27 classes that define the *document type*, roughly oriented on the TANAP main classes. For instance:

- Resolution (Dutch: *Resolutie*)
- Letter (Dutch: *Brief*)
- Minutes (Dutch: *Notulen*)
- ...

On top of these, we introduce the special *Front Matter* as a 28th document type to mark pages that contain text, but are not part of a document, for instance tables of content.

In the document classification task, the page embeddings introduced in Section 2 serve as input for a neural network with a slightly different architecture than the one described in Section 3. Instead of an entire inventory, the input to the bi-directional LSTM layer is now a sub-set of pages that represent a document. The output of the LSTM is passed through a linear layer and a softmax layer to generate a single label for the input.

Data

We use the same datasets as in Section 3. Again, we have manually annotated all the documents in the **primary dataset** with the respective document types. For the **secondary dataset** the document types have been pre-selected as (*Generale Missiven*). For the **tertiary dataset**, we had TANAP document categories available, which we mapped to our document type categorization.

However, the distribution of classes in the dataset dataset remains extremely skewed. Out of the 711 documents that we have sampled for the training data – 6,000 pages in total –, 261 are of the special *Front Matter* type. Among the remaining documents, there are 183 letters, 86 registers, and 41 lists, but only one of type *Invoice* and *Memorandum* respectively. Some other document types are not present at all. In order to get a dataset that is useful for representative experiments, we need to put significant additional effort into annotations, focussing on the underrepresented categories and/or find a trade-off when refining our data model so that it remains meaningful, but makes the dataset machine-learnable.

At this point, we cannot draw empirical conclusions due to an incomplete and skewed dataset, but we take the results shown in Table 2 as an indication that our page embeddings can be used for document classification and other tasks.

Table 2
Document Type Classification: preliminary results on skewed/incomplete dataset

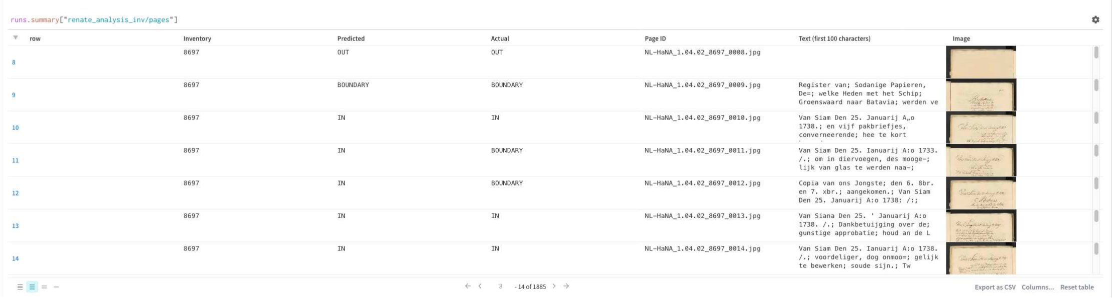
Document Type	F1 Score	Precision	Recall	#Instances
Front Matter	0.92	0.89	0.95	318
Register	0.26	0.26	0.26	105
List	0.26	0.19	0.4	51
Resolution	0.25	0.18	0.4	33
Journal	0.11	0.06	0.57	39
All others (23 types)	0	0	0	344

5. Discussion and Future Work

We present a deep learning approach for extracting documents from a typical historical HTR'd dataset and applied it for the specific, relevant task of document boundary detection. The method is generalizable to outputs from other HTR systems, as well as more broadly to any other related text representations, and for other tasks.

A qualitative analysis of the results on a larger dataset is pending to give practical meaning to the empirically measured precision and recall scores. Due to the ambiguous nature of document boundary definitions, outputs that are not identical with our human annotation could either be incorrect or represent an alternative correct interpretation.

In order to perform a full evaluation, we have set up an evaluation sheet in which multiple human annotators can evaluate their correctness. While empirical results are still lacking, the transparent access to individual results have led to important insights about capabilities and



```
runs.summary["renate_analysis_inv/pages"]
```

row	Inventory	Predicted	Actual	Page ID	Text (first 100 characters)	Image
8	8697	OUT	OUT	NL-HaNA_1.04.02_8697_0008.jpg		
9	8697	BOUNDARY	BOUNDARY	NL-HaNA_1.04.02_8697_0009.jpg	Register van; Sodanige Papieren, De=; welke Heden met het Schip; Groenswaard naar Batavia; werden ve	
10	8697	IN	IN	NL-HaNA_1.04.02_8697_0010.jpg	Van Siam Den 25. Januarij A„o 1738.; en vijf pakbriefjes, converneerende; hee te kort	
11	8697	IN	BOUNDARY	NL-HaNA_1.04.02_8697_0011.jpg	Van Siam Den 25. Ianuarij A:o 1733. /.; om in diervoegen, des mooge-; lijk van glas te werden naa-;	
12	8697	IN	BOUNDARY	NL-HaNA_1.04.02_8697_0012.jpg	Copia van ons Jongste; den 6. 8br. en 7. xbr.; aangekomen.; Van Siam Den 25. Januarij A:o 1738: /:;	
13	8697	IN	IN	NL-HaNA_1.04.02_8697_0013.jpg	Van Siana Den 25. ' Januarij A:o 1738. /.; Dankbetuijging over de; gunstige approbatie; houd an de L	
14	8697	IN	IN	NL-HaNA_1.04.02_8697_0014.jpg	Van Siam Den 25. Ianuarij A:o 1738. /.; voordeliger, dog onmoo=; gelijk te bewerken; soude sijn.; Tw	

Figure 4: We make the individual results visible for transparency and for qualitative analysis.

constraints of quantitative approaches. Figure 4 exemplifies how we display the results per page as logged in Weights & Biases [2].

The unified architecture for creating region and page embeddings opens the door to a variety of tasks, in which these embeddings form the vectorized input and can be fine-tuned per task. As shown, task-specific page embeddings can be used for page sequence tagging (Section 3) and page sequence classification (Section 4).

Other future applications include text quality estimation for targeted post-processing. Previous approaches rely on a mix of human-crafted rules, language-specific dictionaries, and basic machine learning [18, 19]. Page embeddings might make those language-specific rules and resources unnecessary.

Furthermore, our design using stacked neural network layers allows for increasing the number of embedding levels to individual regions lines or even words. In our context, this becomes relevant when segmenting page regions instead of entire pages.

Work in Progress

We want to re-iterate that many aspects of this work are work in progress. Given the practical tasks at hand, however, they always will be so to some extent because task definitions, requirements, and the corresponding data models depend on availability and distribution of data, specific use cases, and interpretation.

We see these dynamics as a given in settings in which computational methods are developed and applied for humanities research that inherently contains a degree of interpretation. We find it important to publish the methodology and implementation along with preliminary results in order to provide a starting point for researchers that have similar, but different challenges.

Acknowledgments

LAHTeR is a project of the Netherlands eScience Center, funded under grant number NLESC.SSIH.2022a.SSIH014.

GLOBALISE is a project of the Huygens Institute with the International Institute of Social History, the Digital Infrastructure Department of the KNAW Humanities Cluster, VU University, the University of Amsterdam, and the Dutch National Archives. It is funded by the Dutch

Research Council (NWO), grant no. 175.2019.003.

References

[1] L. Balk, F. V. Dijk, D. Kortlang, F. Gaastra, H. Niemeijer, and P. Koenders. "The Archives of the Dutch East India Company (VOC) and the Local Institutions in Batavia (Jakarta)". In: *The Archives of the Dutch East India Company (VOC) and the Local Institutions in Batavia (Jakarta).* Brill, 2007. URL: https://brill.com/display/title/14721.

[2] L. Biewald. *Experiment Tracking with Weights and Biases.* 2020. URL: https://www.wandb.com/.

[3] C. M. Bishop. *Pattern recognition and machine learning.* Information science and statistics. New York: Springer, 2006.

[4] P. Delobelle, T. Winters, and B. Berendt. "RobBERT: a Dutch RoBERTa-based Language Model". In: *Findings of the Association for Computational Linguistics: EMNLP 2020.* Ed. by T. Cohn, Y. He, and Y. Liu. Online: Association for Computational Linguistics, 2020, pp. 3255–3265. DOI: 10.18653/v1/2020.findings-emnlp.292. URL: https://aclanthology.org/2020.findings-emnlp.292.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers).* Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423.

[6] A. Guha, A. Alahmadi, D. Samanta, M. Z. Khan, and A. H. Alahmadi. "A Multi-Modal Approach to Digital Document Stream Segmentation for Title Insurance Domain". In: *IEEE Access* 10 (2022), pp. 11341–11353. DOI: 10.1109/access.2022.3144185. URL: https://ieeexplore.ieee.org/document/9684474/.

[7] S. Hochreiter and J. Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735. URL: https://doi.org/10.1162/neco.1997.9.8.1735.

[8] Z. Huang, W. Xu, and K. Yu. *Bidirectional LSTM-CRF Models for Sequence Tagging.* 2015. URL: http://arxiv.org/abs/1508.01991.

[9] P. Kahle, S. Colutto, G. Hackl, and G. Mühlberger. "Transkribus - A Service Platform for Transcription, Recognition and Retrieval of Historical Documents". In: *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR).* Vol. 04. 2017, pp. 19–24. DOI: 10.1109/icdar.2017.307. URL: https://ieeexplore.ieee.org/abstract/document/8270253.

[10] S. Katiyar and S. K. Borgohain. *Image Captioning using Deep Stacked LSTMs, Contextual Word Embeddings and Data Augmentation.* 2021. DOI: 10.48550/arXiv.2102.11237. URL: http://arxiv.org/abs/2102.11237.

[11] L. Keijser. *6000 ground truth of VOC and notarial deeds 3.000.000 HTR of VOC, WIC and notarial deeds*. 2020. DOI: 10.5281/zenodo.6414086. URL: https://zenodo.org/record/6414086.

[12] R. van Koert, S. Klut, T. Koornstra, M. Maas, and L. Peters. “Loghi: An End-to-End Framework for Making Historical Documents Machine-Readable”. In: *Document Analysis and Recognition – ICDAR 2024 Workshops*. Ed. by H. Mouchère and A. Zhu. Cham: Springer Nature Switzerland, 2024, pp. 73–88. DOI: 10.1007/978-3-031-70645-5_6.

[13] E. Manjavacas Arevalo and L. Fonteyn. “Non-Parametric Word Sense Disambiguation for Historical Languages”. In: *Proceedings of the 2nd International Workshop on Natural Language Processing for Digital Humanities*. Taipei, Taiwan: Association for Computational Linguistics, 2022, pp. 123–134. URL: https://aclanthology.org/2022.nlp4dh-1.16.

[14] A. Mao, M. Mohri, and Y. Zhong. *Cross-Entropy Loss Functions: Theoretical Analysis and Applications*. 2023. URL: http://arxiv.org/abs/2304.07288.

[15] T. Mungmeeprued, Y. Ma, N. Mehta, and A. Lipani. “Tab this folder of documents: page stream segmentation of business documents”. In: *Proceedings of the 22nd ACM Symposium on Document Engineering*. San Jose California: Acm, 2022, pp. 1–10. DOI: 10.1145/3558100.3563852. URL: https://dl.acm.org/doi/10.1145/3558100.3563852.

[16] L. A. Ramshaw and M. P. Marcus. *Text Chunking using Transformation-Based Learning*. 1995. DOI: 10.48550/arXiv.cmp-lg/9505040. URL: http://arxiv.org/abs/cmp-lg/9505040.

[17] N. Reimers and I. Gurevych. “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 3980–3990. DOI: 10.18653/v1/D19-1410. URL: https://www.aclweb.org/anthology/D19-1410.

[18] P. Schneider and Y. Maurer. “Rerunning OCR: A Machine Learning Approach to Quality Assessment and Enhancement Prediction”. In: *Journal of Data Mining & Digital Humanities* 2022.Digital humanities in languages (2022). DOI: 10.46298/jdmdh.8561. URL: https://jdmdh.episciences.org/10239.

[19] C. Schnober. *text_quality*. Version 0.3.1. 2023. DOI: 10.5281/zenodo.8189892. URL: https://www.github.com/laHTeR/htr-quality-classifier.

[20] M. Schuster and K. Paliwal. “Bidirectional recurrent neural networks”. In: *Signal Processing, IEEE Transactions on* 45 (1997), pp. 2673–2681. DOI: 10.1109/78.650093.

[21] R. Smit. *Reusing traditional finding aids for the GLOBALISE infrastructure*. 2024. URL: https://globalise.huygens.knaw.nl/from-abc-to-voc-volume-utilizing-traditional-finding-aids-for-the-globalise-infrastructure/.

[22] *VOC transcriptions v2 - GLOBALISE*. Version V1. 2024. DOI: 10622/lvxsbw. URL: https://hdl.handle.net/10622/LVXSBW.

[23] U. Yaseen and S. Langer. *Neural Text Classification and Stacked Heterogeneous Embeddings for Named Entity Recognition in SMM4H 2021*. 2021. DOI: 10.48550/arXiv.2106.05823. URL: http://arxiv.org/abs/2106.05823.

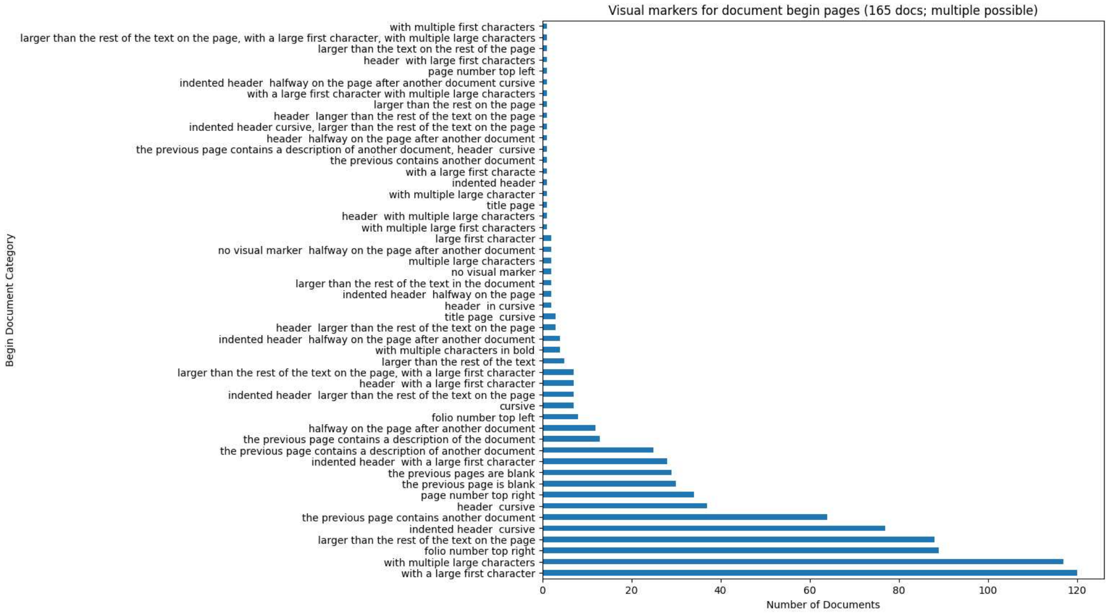


Figure 5: Visual markers that indicate document begin pages, based on manual analyses.

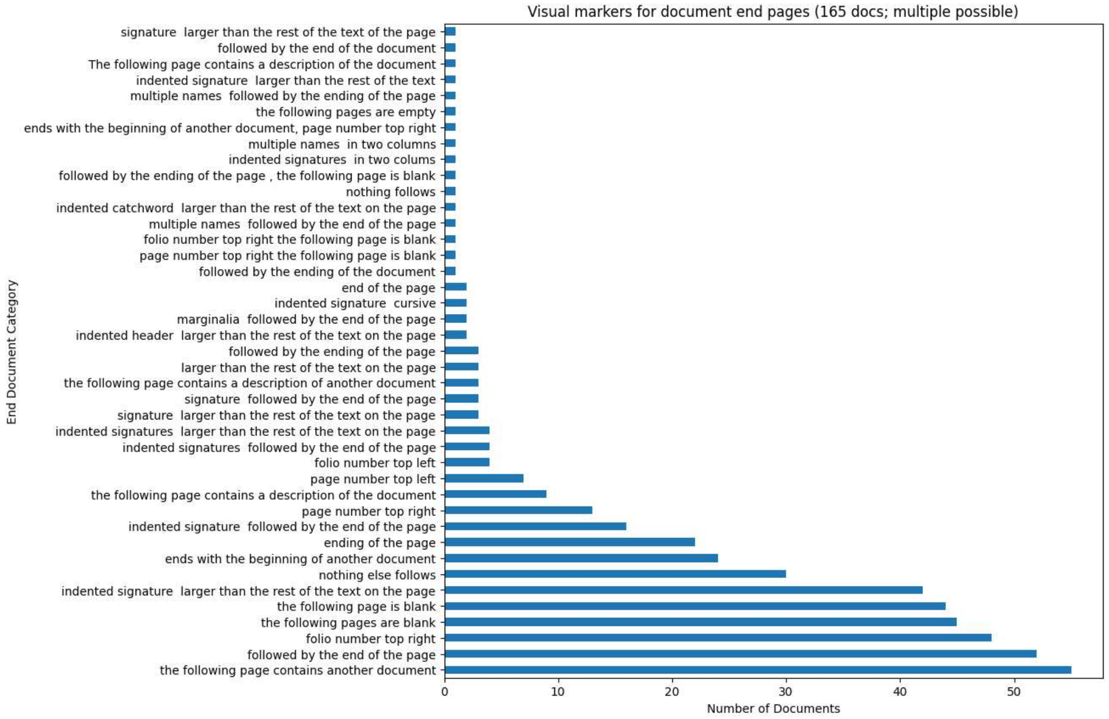


Figure 6: Visual markers that indicate document end pages, based on manual analyses.