Scalable handwritten text recognition system for lexicographic sources of under-resourced languages and alphabets

Jan Idziak[1], Artjoms Šeļa[2], Michał Woźniak[2], Albert Leśniak[2], Joanna Byszuk[2], and Maciej Eder[2]

[1] independent scholar, Singapore
jan.idziak@gmail.com
[2] Institute of Polish Language, Polish Academy of Sciences,
al. Mickiewicza 31, 31-120 Krakow, Poland
{artjoms.sela,michal.wozniak,albert.lesniak,
joanna.byszuk,maciej.eder}@ijp.pan.pl

Abstract. The paper discusses an approach to decipher large collections of handwritten index cards of historical dictionaries. Our study provides a working solution that reads the cards, and links their lemmas to a searchable list of dictionary entries, for a large historical dictionary entitled the *Dictionary of the 17th- and 18th-century Polish*, which comprizes 2.8 million index cards. We apply a tailored handwritten text recognition (HTR) solution that involves (1) an optimized detection model; (2) a recognition model to decipher the handwritten content, designed as a spatial transformer network (STN) followed by convolutional neural network (RCNN) with a connectionist temporal classification layer (CTC), trained using a synthetic set of 500,000 generated Polish words of different length; (3) a post-processing step using constrained Word Beam Search (WBC): the predictions were matched against a list of dictionary entries known in advance. Our model achieved the accuracy of 0.881 on the word level, which outperforms the base RCNN model. Within this study we produced a set of 20,000 manually annotated index cards that can be used for future benchmarks and transfer learning HTR applications.

Keywords: Handwritten Text Recognition · index cards archives · lexicography · Neural Network · Convolutional Neural Network · Recurrent Neural Network · Connectionist Temporal Classification · keras ocr · ResNet · Spatial Transformer Networks · synthetic dataset.

1 Introduction

Decades of lexicographic work that was done before the popularization of machine-readable texts provide rich lexicographic and/or linguistic data that is extremely hard to reuse today or to be integrated into modern databases, corpora and collections. Not only are these original resources handwritten, but they are also unstructured, or at best their structure is limited to an alphabetical order of

the respective items. Card files served as tools of lexicographic description and, when collected into catalogues, allowed random access to vast bodies of lexical information. These cards were building blocks of lexicons and dictionaries, long before corpus linguistics that relied on digitized texts appeared [18]. The lexicographic resources in question involve millions of handwritten cards for various historical dictionaries, ranging from Latin (with the archetypical *Thesaurus Linguae Latinae*, one of the first initiatives of this kind), to medieval and modern language varieties (Middle Dutch, Old Czech, Old Norse Prose, or Middle High German to name but a few). In most cases, the index cards are acquired throughout several decades – sometimes dating back to the 19$^{\text{th}}$ century – and they contain comprehensive documentation for all known words of respective language varieties.

The lexicographic collections held at the Institute of Polish Language of the Polish Academy of Sciences are no exception, with its extensive card catalogues of *The Old Polish Dictionary*, *The Dictionary of the 17th- and 18th-century Polish*, *The Dictionary of Polish Dialects*, *The Great Dictionary of Polish*, as well as a few onomastic dictionaries of proper nouns. A single index card contains a lemma (a base word in a header) followed by its context excerpted from actual historical documents. Stored in dedicated boxes and alphabetized, the index cards are used by lexicographers to compose subsequent dictionary entries. Since most of the aforementioned dictionaries are not completed yet, the handwritten index cards serve as a work-in-progress source of information, and, even in the case of the dictionaries that are already published, the index cards are still valid as their supplementary materials. The biggest challenge, however, is that they are not machine-readable, and not linked to the searchable lexicographic databases.

While recent years saw a development of numerous approaches to optical and handwritten text recognition (HTR) also in the relation to humanistic data, e.g. Transkribus [16,20] which provides excellent performance, such solutions are better suited to longer texts, fewer scribal hands and require significant amounts of training data [8], also posing limitations as to the number of pages that can be annotated. Meanwhile, the index cards contain short excerpts, typically no more than a sentence of context next to the lemma, followed by source description, and are produced by numerous lexicographers, often showing inconsistent handwriting style that can be understood only by themselves or other team members. In fact, while the second poses a significant challenge, also in the case of the need to prepare manually annotated training set, computer vision methods which rely on less easy to observe patterns hold great promise of perhaps outperforming human reading of more illegible scribblings.

This project sets up an operational workflow for retrieving lexical data from handwritten card catalogues, followed by matching their lemmas to the list of dictionary entries. To build and test a prototype, we have chosen *The Dictionary of the 17th- and 18th-century Polish*, which is a good example of a lexicographic source that combines traditional materials of 2.8 million handwritten index cards with modern technologies – the dictionary itself is a fully digital database, and it is linked to an annotated corpus of the 17$^{\text{th}}$-century Polish [21,4]. Moreover,

in a pilot study a small selection of the index cards was manually mapped onto a list of dictionary entries [3]. This preliminary work, however, clearly shows that manual mapping is hardly feasible in real-scale setups and would involve an immense effort expressed in thousands of working hours. Our project aims at overcoming this limitation by using an automated approach that would simplify the work of lexicographers in preparing the digital entries. The main goal of the project, however, goes beyond the prototype applied to the 17^{th}-century lexicographic sources. The diverse range of the obtained outcomes makes this study potentially interesting both from a computer sciences point of view, as well as from a digital humanities perspective.

The research presented in this paper provides the following contributions:

1. We propose a unified modular workflow that is adjustable to any language, since the model relies solely on a synthetic dataset; our workflow can be easily extended to other under-resourced languages, including languages with extended Latin alphabets or non-Latin scripts.
2. We provide a working HTR detection and recognition prototype (also as a deployed demo web application) that outperforms baseline models significantly.
3. We provide a synthetic dataset of artificially generated 500,000 words in Polish, supplemented by another set of 30,000 random strings with uniform distribution of Polish diacritics.
4. We offer a manually labelled set of 20,000 words in Polish to be used as ground truth in future applications and model evaluation settings.

2 Related work

Modern HTR heavily leans towards solutions based on Artificial Neural Networks and it was shown that various architectures of deep learning improve performance in the handwriting recognition task [10,22,27,12,15,9,6,29,28] compared to Hidden Markov models [25,26]. Handwritten text imposes severe challenges for a machine vision technologies that we counter by combining several known methods: (1) the problem of significant variation in writing style, shapes, sizes and possible deformations of characters is solved through spatial deformation rectification achieved with the Spatial Transformation Network that learns translation-invariant features [15]. (2) Since source material in HTR tasks is often based on a large and diverse lexicon which adds to the difficulty of decision making of the model, Word Beam Search is often used to decode CTC layer and constrain prediction errors [24]. Many best-performing post-OCR error correction systems depend on contextual awareness provided by a language model [23], which cannot be adapted to our case of isolated index words. Instead we rely on lexicon-aided decoding of the model's predictions that broadly follows the character-level error correction framework [7]. (3) It is often very difficult to achieve a system generalizability in HTR in a specific domain because of the lack of a large amount of ground truth handwriting samples. Recent studies propose to compensate this by generating vast amounts of synthetic images from character strings [14]. It was

even shown that CNN algorithm trained only on the synthetic data outperformed other methods on the text detection task [12]. (4) Finally, the level at which text segmentation for HTR is deployed was subjected to discussion. Early approaches [19] employed line segmentation in the HTR context, while word segmentation or character region awareness methods [12,13] are gaining more popularity recently.

3 Data

3.1 Original data

Our main focus was the *The Dictionary of the 17th- and 18th-century Polish* (`https://sxvii.pl/`), a partially-completed lexicographic database that is based on roughly 2.8 million index cards manually filled by different hands (in rare cases typewritten) in the years 1954–1995 [4,3]. The cards are stored in 836 alphabetized boxes, and, after a digitization project conducted in the years 2010–2015, they are now freely accessible in bitmap image format, saved under names linking to their respective boxes (e.g. *Egzekucja–Ekspediowanie*). For the sake of this study, we drew a sample of 100,000 cards taken uniformly across all boxes, which made up our primary dataset (denoted as `PL-100k-main` hereafter).

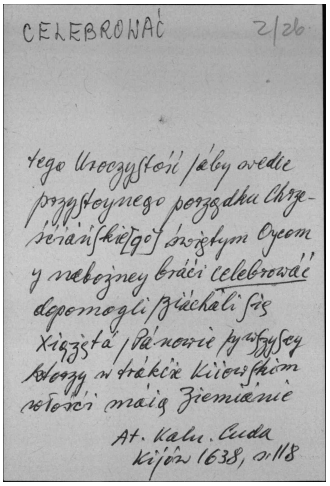

Fig. 1. A handwritten index card from *The Dictionary of the 17th- and 18th-century Polish*.

Almost all of the cards – rare exceptions being phonetic variant pointers – followed a conventional scheme for encoding a given word's attestation (see Fig. 1): (1) a lemma served as an index which ensured navigation and accessibility, while (2) the body of the card registered immediate context of the word occurrence, followed

by (3) a bibliographical reference to sources and sometimes including other information (grammatical form, alternate spelling, etc.). Our lemma detection procedure at times encountered difficulties when the additional information about grammatical form was written next to the header lemma, rather than within the card's body. A significant part of the cards from the *The Dictionary of the 17th- and 18th-century Polish* had their index word written in "handwritten capitals" (majuscule alphabet) to keep the words recognizable, which limits the variability of shapes and maintains somewhat clear boundaries between letters. A mix of minuscule and all-majuscule handwriting meant that the HTR model should be able to perform simultaneously on both of these modes of writing, which added an additional step to the transfer learning workflow (as discussed below).

The list of all the 86,000 dictionary entries of the *The Dictionary of the 17th- and 18th-century Polish* is publicly accessible (`https://sxvii.pl/`). We used the list to serve as a constrained set of possible words to match them against the resulting predictions from our HTR system.

3.2 Synthetic data for training

The majority of already existing training datasets for HTR are limited to English. Since Polish language uses an extended Latin script with additional 16 diacritics (8 lowercase and 8 capitalized), we had to make sure that Polish is at least partially represented [11]. Our transfer-learning approach involved the already existing large dataset of English words CVIT [17] that we further enhanced with an artificially generated set of Polish words. To this end, we randomly excerpted 500,000 actual Polish words from the corpus of 17$^{\text{th}}$-century Polish texts *Korba* [21], and generated their bitmap representations using a variety of fonts that mimicked the handwriting both in lowercase and uppercase (this is our `PL-500k-synthetic` dataset). Not only were the words and the font shapes picked at random, but also the final bitmap images were slightly distorted using different augmentation steps. The augmentation distortion included (1) posterization – which maximizes the image contrast, (2) equalization of the image histogram, (3) solarization – which inverts all pixel values above a threshold, (4) affine geometric transformation. Consequently, the set we obtained consisted of artificial bitmap representations of actual Polish words. This allowed for training the representation of the Polish diacritics, while the proportion of diacritics to standard characters followed their natural distribution.

Apart from the above `PL-500k-synthetic` dataset, we also prepared an additional set of 30,000 randomly generated strings containing solely Polish diacritical marks – with uniform distribution – in order to improve performance for rare polish diacritics. We denote this set as `PL-30k-diacritics`.

3.3 Manually labelled subsets

To obtain a carefully curated set for evaluation purposes, we drew a small sample from the original dataset, namely 20,000 bitmap images, that were then manually corrected for their bounding boxes, and manually labeled by the project

members. This set (`PL-20k-hand-labelled`) was our primary evaluation set, since it provided ground truth for both neatly cropped images and corrected labels. The set has been made publicly available for future benchmarking and applications. In order to facilitate the tedious annotation work, we applied the annotation tool Prodigy (`https://prodi.gy/`).

Yet another manually annotated subset was prepared (`PL-3k-boundaries`) to optimize the performance of word detection. We drew at random 3,000 images from the original dataset, applied the word detection module as discussed above, and manually-corrected the resulting bounding boxes around the detected lemmas.

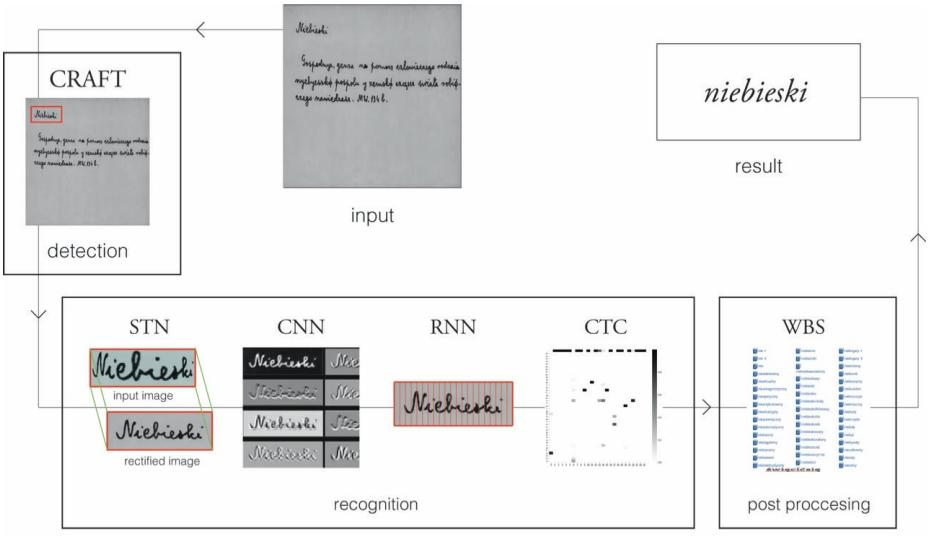

Fig. 2. The HTR workflow, including the detection, recognition, and post-processing steps.

4 Methodological workflow

Our general workflow involved three main parts: (1) detection, (2) recognition, and (3) postprocessing. The first two steps are based on deep learning and convolutions, while the third step is an optimization technique aimed at improving the final predictions of the neural network.

4.1 Detection of the index word

Since all cards had a header lemma, usually placed distinctly from the body text, it is relatively easy to access it: we cut a card to top 300 pixels in order to optimize for computation time, and used Keras OCR Craft model for text

detection and bounding box assignment [2]. Then for each card, we identified the first bounding box located in the top-left area, which in the vast majority of cases contains the word in question. We selected 3,000 hand-labeled images to optimize the position of the bounding boxes (the `PL-3k-boundaries` set).

4.2 Recognition

The recognition stage can be broken down to four consecutive components based on TPS-ResNet-BiLSTM-CTC architecture proposed by Baek at al. [1]. We used already existing pre-trained model that we further improved:

1. Input text image was first rectified with the help of the Spatial Transformer Network, or STN [15] with Thin Plate Spline (TPS) transformation [29]. The aim of this step is to ensure that the images are consistent in terms of contrast, saturation and so forth, and thus easier to process at the feature extraction step.
2. Feature extraction using a Convolutional Neural Network (CNN) setup. It extracted relevant features from the image and focused on attributes that are characteristic to particular characters. After a few preliminary tests, we chose a ResNet backbone, because it provided a clear improvement in accuracy.
3. The features extracted in step 2 are fed sequentially into the Bidirectional LSTM layer (BiLSTM).
4. Finally, the Connectionist Temporal Classification (CTC) layer was involved. The benefit of using it is at least two-fold. Firstly, the predictions show the ability to overcome a variable size of the input sequence, even if the number of features is fixed. Secondly, the CTC layer accounts for words with repeated letters, thus helping to differentiate between, say, the words "to" and "too". Also, because most of the publicly available datasets used English, a standard Attention layer would not guarantee sufficiently good performance for other languages. The CTC layer, on the contrary, provided a matrix of all possible predictions which could be further generalized beyond English.

4.3 Transfer learning

In our approach we first took an existing model pre-trained on two datasets: MJSynth (MJ) [14] and SynthText (ST) [15]. These datasets are not designed for HTR problems and do not provide Polish characters, therefore we involved a few additional datasets to enhance the model. Firstly, we added the CVIT database with 9 million images of handwritten words based on English corpus. Subsequently, further domain improvement was done using Polish words from a synthetic dataset `PL-500k-synthetic` as discussed above. The last step involved enhancing the performance specifically for the recognition of Polish diacritics. To achieve this, we used the randomly generated strings containing all characters with uniform distribution (`PL-30k-diacritics`).

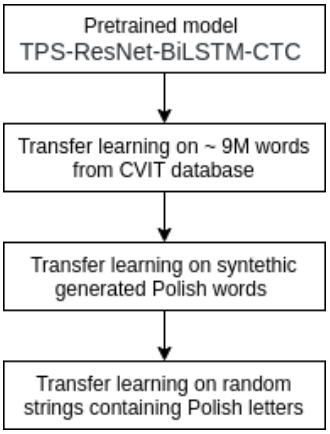

Fig. 3. Transfer learning workflow.

Initially, we also used the IAM handwritten database containing carefully-labelled handwritten words [19]. After several rounds of transfer learning tests, however, it became clear that the dataset representation and the sample images significantly differ from texts produced in natural conditions. This is caused by the fact that the words in the IAM handwritten database are cut closely around the word outlines, which confuses the recognition system. Our observations suggest that using this dataset leads to overfitting of neural network models with a large number of parameters. Consequently, any applications based on word representation of the IAM handwritten dataset do not seem suitable in real life situations. We would even argue that this dataset should not be used as the main dataset for performance evaluation as well.

4.4 Postprocessing

The predictions as produced by our workflow will always contain some wrongly reconstructed words. However, some of these mistakes are relatively easy to correct, due to the non-random nature of the language. For instance, a human would instinctively guess that the string "cvolution" resembles the word "evolution". The string "ancl" would require an additional split second to decipher, because it could be reconstructed as "and", as "ant" and perhaps even as "uncle" or "anele". Since the list of possible words is limited, an optimization algorithm can be used that matches the input sequence with the nearest element from the closed set of words known in advance. Real-life situations might be more challenging, given the fact that new words emerge at times, e.g. the algorithm would disregard the string "covfefe" as a valid word, and would replace it either with "coffee" or with "coverage".

In the case of our project, however, a vast majority of words written down on index cards match the close set of words stored as a list of 86,000 dictionary entries.

Additionally, we took advantage of the fact that the index cards are alphabetized (as is the list of dictionary entries), and stored in 836 boxes with known ranges of the alphabet ascribed to each box. Consequently, in our constrained Word Beam Search (WBC) approach not only did we take into consideration the CTC match of the predictions and the expected dictionary entries, but we also assumed that it is very unlikely to match an index card from a given box to a word belonging to a distant part of the alphabetized list.

In the first round of our procedure, we additionally involved a quality check step. To this end, we prepared the aforementioned `PL-20k-hand-labelled` dataset: we randomly selected 20,000 index cards, manually checked the predictions, and corrected all the misrecognized words. The aim of this step was two-fold: firstly, to allow for re-training the model with the clean ground-truth dataset (while keeping in mind that for the sake of this study we didn't apply it, for the sake of a clear-cut separation of the training set and the evaluation set) and secondly, to prepare 20,000 manually corrected cards for public access for future benchmarking and improving HTR models.

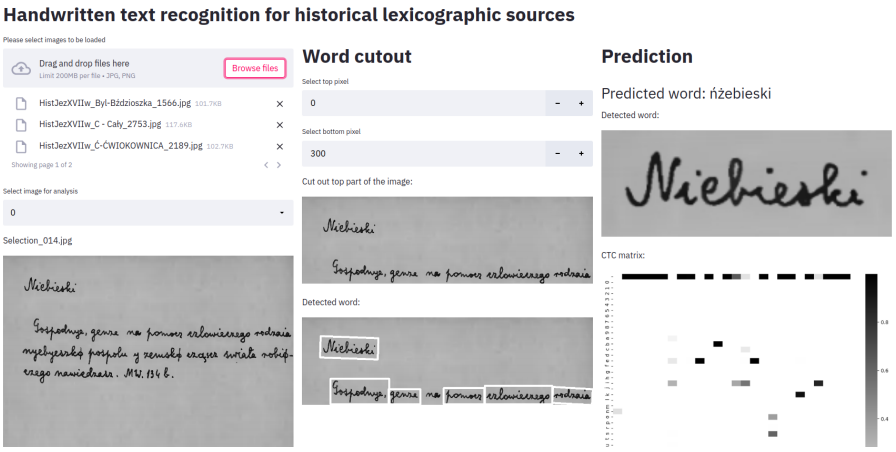

Fig. 4. Online demo application of the HTR system.

5 Application

Alongside our full-size system that is capable of consecutively processing millions of cards, we also designed an online application for testing our workflow and visually presenting how a given index card is being processed. After the file is selected, the app automatically performs cutting, detection and recognition of the word. In post-processing, the app optimizes for the best path CTC encoding. Both vanilla prediction and the CTC matrix are available for inspection.

6 Results

The results of the system are based on accuracy scores achieved on a subset of the original data (the `PL-100k-main` datset). The detection component of the workflow achieved 0.93 of intersection over union on 3,000 hand labeled images (the `PL-3k-boundries` dataset). The results for the recognition model are best presented in comparison to a baseline model. We have excluded 20,000 index cards (the `PL-20k-manual`) to serve as a test set, the average word length was 5.99 characters. Accuracy is calculated as the number of words that were classified correctly divided by the number of all words in the dataset:

$$Acc = \frac{1}{n}\sum_{i}^{n} \mathbb{1}_{y_i=\hat{y}_i}$$

We use Levenshtein distance as the edit distance. Average edit distance is calculated as the sum of the edit distance divided by the number of words. Average normalized edit distance is calculated as averaged edit distance per number of characters in the word:

$$\Delta_{Lnorm}(i) = \frac{\Delta_L(y_i, \hat{y}_i)}{\max(\text{length}(y_i), \text{length}(\hat{y}_i))}$$

We compared the performance of three decoding approaches of the CTC layer in the RCNN model: (1) best path decoding, (2) word beam search, and (3) constrained word beam search. The results are shown in Table 1. As can be observed, we achieved a significant improvement over the base model offered by an RCNN (TPS-ResNet-BiLSTM-CTC) model with no transfer learning and no CTC and Word Beam Search refinement.

Table 1. Results achieved by the HTR models. All of them are based on TPS-ResNet-BiLSTM-CTC architecture. BP – best path, WBS – word beam search, WBS-C – constrained word beam search, POL – model trained on the Polish synthetic set.

Model	Word accuracy	Normalised edit distance	Edit distance	Average edit on misclassified
BP	0.3755	0.1898	1.0871	**1.7484**
WBS	0.0995	0.6194	6.1763	6.8711
BP-POL	0.4332	0.2246	1.2120	2.0945
WBS-POL	0.6655	0.2033	1.0993	2.7063
WBS-C-POL	**0.8810**	**0.0479**	**0.3165**	3.2125

Since our pipeline is, among other things, aimed to aid lexicographers in linking index cards to existing databases, the model that minimizes edit distance on *wrong* predictions sometimes could be more useful than the WBS model that is designed for word-level accuracy and makes use of external information (alphabet range of a box). Average edit distance on wrongly recognized words

(Table 1) shows this effect: both best path decoding models have less edits than WBS decoding (1.75 and 2.09), suggesting that in a realistic setting of manual work with the full-scale collection, less greedy models could provide more "useful" predictions despite the drop in word-level accuracy. In addition, the best-performing constrained WBS model has a higher rate of edits in misclassified words than unconstrained WBS which implies that sometimes words fall out of the box's alphabetical range but then are still forcefully fitted to that range by WBS-C. Most probably, these mistakes happen because index word position could be recognized incorrectly in a situation when a card has multiple words on the top.

7 Discussion

Our results show a trade-off between a straightforward word-level accuracy and amount of noise captured by a model. High values of average edit distance on the misclassified words for the Word Beam Search models show that the algorithm is likely to pick up noise in the activation map and predict long words even though the best path encoding would ignore such activation. If the label is "a" and the predicted word is "aproksymacja", the edit distance would be eleven. This is also visible for the vanilla base model with the Word Beam Search encoding: it achieves the accuracy of less than 0.1, while the base model is able to get to 0.37. Our best-performing model achieves the word-level accuracy of 0.88 due to highly constrained output, limited by the alphabetical range of a box for a source word, but at the same time its tendency to aggressively fit predictions to longer or unrelated words remains an issue.

The vanilla RCNN model had lower edit distances than the vanilla model with the knowledge of the full Polish alphabet. This happened because Polish diacritics extend possibilities of decoding and are simultaneously not frequently encountered. Thus, a model that is aware of Polish diacritics, could predict *a* as *ą* or *e* as *ę*, while in real life *ę* and *ą* are infrequent. At the same time, the vanilla base model would not make these errors as it does not have *ą* or *ę* available for a prediction at all.

Our work highlights the importance and possibilities of automated information extraction from a historical archive, on the example of index card catalogues of dictionaries of historical variants of Polish developed in the Institute of Polish Language of the Polish Academy of Sciences. The way in which index cards are organized, facilitates the recovery of some parts of the source structure (e.g. index lemma, body, references) and invites further processing, such as knowledge linking, that goes beyond the undiscriminating plain text recognition from a given image. That recognition of already structured information could be further extended with additional layers of layout analysis of a card image.

8 Conclusion

In this paper, we presented a HTR solution tailored for processing large collections of handwritten index card catalogues. Although primarily designed to deal with

the Polish language and lexicographic sources, our solution also expands HTR applicability to under-resourced languages and alphabets, providing domain-specific dataset that could be further reused for a wide array of tasks. The linguistic archives around the world present a wide variety of historical, dialectal, onomastic and other lexicographic data. Such linguistic projects were often undertaken decades before the creation of corpus linguistic tools and digital databases. HTR pipelines could contribute greatly to quickening the pace of work on turning index cards into dictionaries. In the future, a similar approach could be used to globally solve the problem of linking data of the past to the existing resources and linguistic platforms which today massively inhabit the digital space and utilize its affordances. The recognition of different handwriting and definition building patterns can also serve as an invaluable resource for examining the conventions of work in dictionary project as well as the study of individual contributions.

Software and data

Code, models and data: `https://github.com/perechen/htr_lexicography`
Web demo application: `http://149.156.30.114:8503/`

The elements of the system implemented in (or inspired by) other solutions: synthetic data generation [5], detection model [2], recognition model [1], CTC and Word Beam Search [24].

The 17th-century index cards (the original dataset): `https://rcin.org.pl/dlibra/publication/20029`.

Acknowledgements

This research was partly conducted as a result of a project supported by Poland's National Science Centre (project number UMO-2013/11/B/HS2/02795). The authors are grateful to Bartłomiej Borek for his IT support, which included setting up the server and the environment for conducting our experiments.

References

1. Baek, J., Kim, G., Lee, J., Park, S., Han, D., Yun, S., Oh, S.J., Lee, H.: What is wrong with scene text recognition model comparisons? dataset and model analysis. In: International Conference on Computer Vision (ICCV) (2019)
2. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9365–9374 (2019)
3. Bilińska-Brynk, J., Rodek, E.: Paper quotation slips to the Electronic Dictionary of the 17th-and 18th-Century Polish – digital index and its integration with the Dictionary. In: EURALEX XIX Proceedings. pp. 465–470 (2020)
4. Bronikowska, R., Majdak, M., Wieczorek, A., Żółtak, M.: The Electronic Dictionary of the 17th-and 18th-century Polish – towards the open formula asset of the historical vocabulary. In: EURALEX XIX Proceedings. pp. 471–475 (2020)

5. Chu, W.: Text renderer. https://github.com/Sanster/text_renderer (2021)
6. Doetsch, P., Kozielski, M., Ney, H.: Fast and robust training of Recurrent Neural Networks for offline handwriting recognition. In: 2014 14th International Conference on Frontiers in Handwriting Recognition. pp. 279–284 (2014). https://doi.org/10.1109/ICFHR.2014.54
7. Farra, N., Tomeh, N., Rozovskaya, A., Habash, N.: Generalized character-level spelling error correction. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. vol. 2, pp. 161–167. Association for Computational Linguistics, Baltimore, Maryland (2014). https://doi.org/10.3115/v1/P14-2027, http://aclweb.org/anthology/P14-2027
8. Franzini, G., Kestemont, M., Rotari, G., Jander, M., Ochab, J.K., Franzini, E., Byszuk, J., Rybicki, J.: Attributing authorship in the noisy digitized correspondence of Jacob and Wilhelm Grimm. Frontiers in Digital Humanities **5** (2018). https://doi.org/10.3389/fdigh.2018.00004
9. Graves, A., Fernández, S., Schmidhuber, J.: Multi-dimensional Recurrent Neural Networks. In: de Sá, J.M., Alexandre, L.A., Duch, W., Mandic, D. (eds.) Artificial Neural Networks – ICANN 2007. pp. 549–558. Lecture Notes in Computer Science, Springer (2007). https://doi.org/10.1007/978-3-540-74690-4_56
10. Graves, A., Schmidhuber, J.: Offline handwriting recognition with multidimensional Recurrent Neural Networks. Advances in Neural Information Processing Systems **21**, 545–552 (2008), https://proceedings.neurips.cc/paper/2008/hash/66368270ffd51418ec58bd793f2d9b1b-Abstract.html
11. Grzelak, D., Podlaski, K., Wiatrowski, G.: Analyze the effectiveness of an algorithm for identifying Polish characters in handwriting based on neural machine learning technologies. Journal of King Saud University – Computer and Information Sciences (2019). https://doi.org/https://doi.org/10.1016/j.jksuci.2019.08.001
12. Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2315–2324 (2016), https://openaccess.thecvf.com/content_cvpr_2016/html/Gupta_Synthetic_Data_for_CVPR_2016_paper.html
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015), http://arxiv.org/abs/1512.03385
14. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition (2014), http://arxiv.org/abs/1406.2227
15. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial Transformer Networks (2016), http://arxiv.org/abs/1506.02025
16. Kahle, P., Colutto, S., Hackl, G., Mühlberger, G.: 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, pp. 19–24 (2007). https://doi.org/10.1109/ICDAR.2017.307
17. Krishnan, P., Jawahar, C.: Matching handwritten document images. In: European Conference on Computer Vision. Springer (2016)
18. Landau, S.I.: Dictionaries: The art and craft of lexicography. Cambridge University Press, 2 edn. (2001)
19. Marti, U.V., Bunke, H.: The IAM-database: An English sentence database for offline handwriting recognition. International Journal on Document Analysis and Recognition **5**(1), 39–46 (2002). https://doi.org/10.1007/s100320200071
20. Muehlberger, G., Seaward, L., Terras, M., Ares Oliveira, S., Bosch, V., Bryan, M., Colutto, S., Déjean, H., Diem, M., Fiel, S., Gatos, B., Greinoecker, A., Grüning,

T., Hackl, G., Haukkovaara, V., Heyer, G., Hirvonen, L., Hodel, T., Jokinen, M., Kahle, P., Kallio, M., Kaplan, F., Kleber, F., Labahn, R., Lang, E.M., Laube, S., Leifert, G., Louloudis, G., McNicholl, R., Meunier, J.L., Michael, J., Mühlbauer, E., Philipp, N., Pratikakis, I., Puigcerver Pérez, J., Putz, H., Retsinas, G., Romero, V., Sablatnig, R., Sánchez, J.A., Schofield, P., Sfikas, G., Sieber, C., Stamatopoulos, N., Strauß, T., Terbul, T., Toselli, A.H., Ulreich, B., Villegas, M., Vidal, E., Walcher, J., Weidemann, M., Wurster, H., Zagoris, K.: Transforming scholarship in the archives through handwritten text recognition: Transkribus as a case study. Journal of Documentation **75**(5), 954–976 (2019). https://doi.org/10.1108/JD-07-2018-0114

21. Ogrodniczuk, M., Gruszczyński, W.: Connecting data for digital libraries: The library, the dictionary and the corpus. In: Jatowt, A., Maeda, A., Syn, S.Y. (eds.) Digital Libraries at the Crossroads of Digital Information for the Future. pp. 125–138. Lecture Notes in Computer Science, Springer International Publishing (2019). https://doi.org/10.1007/978-3-030-34058-2_13
22. Pal, A., Singh, D.: Handwritten English character recognition using neural network. International Journal of Computer Science & Communication **1**(2), 141–144 (2010)
23. Rigaud, C., Doucet, A., Coustaty, M., Moreux, J.P.: ICDAR 2019 Competition on Post-OCR Text Correction. 15th International Conference on Document Analysis and Recognition pp. 1588–1593 (2019), `https://hal.archives-ouvertes.fr/hal-02304334/document`
24. Scheidl, H., Fiel, S., Sablatnig, R.: Word Beam Search: A Connectionist Temporal Classification decoding algorithm. In: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 253–258 (2018). https://doi.org/10.1109/ICFHR-2018.2018.00052
25. Sánchez, J.A., Romero, V., Toselli, A.H., Vidal, E.: Icfhr2014 competition on handwritten text recognition on Transcriptorium datasets (HTRtS). In: 2014 14th International Conference on Frontiers in Handwriting Recognition. pp. 785–790 (2014). https://doi.org/10.1109/ICFHR.2014.137
26. Sánchez, J.A., Romero, V., Toselli, A.H., Vidal, E.: Icfhr2016 competition on handwritten text recognition on the READ dataset. In: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 630–635 (2016). https://doi.org/10.1109/ICFHR.2016.0120
27. Voigtlaender, P., Doetsch, P., Ney, H.: Handwriting recognition with large multidimensional Long Short-Term Memory Recurrent Neural Networks. In: 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 228–233 (2016). https://doi.org/10.1109/ICFHR.2016.0052
28. Xiao, S., Peng, L., Yan, R., Wang, S.: Deep network with pixel-level rectification and robust training for handwriting recognition. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 9–16 (2019). https://doi.org/10.1109/ICDAR.2019.00012
29. Yin, X., Yin, X., Huang, K., Hao, H.: Robust text detection in natural scene images. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(5), 970–983 (2014). https://doi.org/10.1109/TPAMI.2013.182