# Building and Improving an OCR Classifier for Republican Chinese Newspaper Text

## Arnold, Matthias

arnold@uni-heidelberg.de
Heidelberg Centre for Transcultural Studies, Universität Heidelberg, Germany

## Henke, Konstantin

konstantin.henke@protonmail.ch
Institut für Computerlinguistik, Universität Heidelberg, Germany

For more than a decade, Republican magazines and newspapers have been collected by institutes and projects now joined in the Centre for Asian and Transcultural Studies (CATS) at Heidelberg University. Our platform “Early Chinese Periodicals Online” (ECPO, https://uni-heidelberg.de/ecpo), provides open access to more than 300.000 digital images and their metadata, cf. Arnold and Hessel (2020) . Since the material consists mostly of image scans, the project ran a number of experiments to explore possible approaches towards full text generation (Arnold, 2021). For newspapers printed in Latin scripts much has changed since Rose Holley commented item “Use the ‘training’ facility (artificial intelligence) in the OCR software” with “Not viable for cost effective mass scale digitization” and noted “Do not pursue” in her list of “Potential methods of improving OCR accuracy suggested by ANDP team” (Holley, 2009, table 2, item 9). Today, when researchers write that “transforming [historical newspapers] into machine-readable data by means of OCR poses some major challenges” they do that while they introduce their own OCR pipeline (Holley, 2009).

Unfortunately, these approaches cannot just be adopted to historical Chinese newspapers. As we have shown, especially complex layout and resulting difficulties of reliable automatic page segmentation have so far prevented full text generation of these newspapers even within China (Arnold, 2021; Arnold, forthcoming; Arnold et al., forthcoming). In this long abstract we present the first results from a systematic approach towards full text extraction from a Republican China newspaper ( 1). Our basis is a small corpus for which text ground truth exists. We present our character segmentation method which produces about 90.000 images of characters. Based on the hypothesis that pre-training on extensive amounts of suitably augmented character images will increase the OCR accuracy for evaluation on real-life character image data, we generate synthetic training data. We then compare the OCR recognition results and show that a combination of synthetic and real characters produces best results. Finally, we propose a method that makes use of a masked language model for OCR error correction.

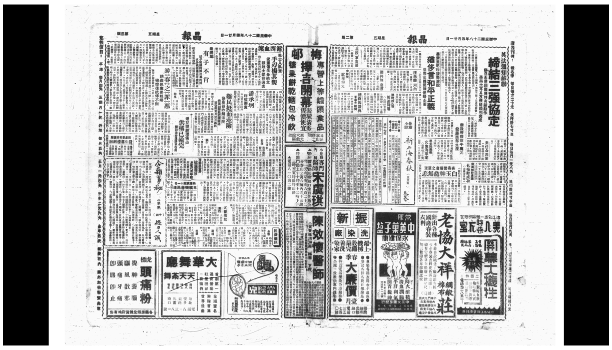
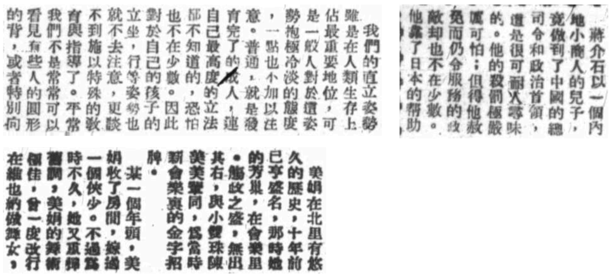


Fig. 1: An example fold from *Jing bao* 晶報 *(The Crystal)*, April 21, 1939, pages 2-3.

Note: We will treat single rectangular text blocks (Fig. 2) as given and proceed from here to present effective methods for generating a data set later used to train an OCR model. We show that pre-training on artificially created training data can significantly improve OCR accuracy. Due to the limited scope of the presented experiments, this approach is still limited in terms of retrieved glyph size, image quality and font style, hence the model is not necessarily directly applicable to other historical Chinese documents.



我們的直立姿勢雖是在人類生存上佔最重要地位，可是一般人對於這姿勢抱極冷淡的態度，一點也不加以注意。普通，就是發育完了的成人，連自己最高度的立法卻不知道的，恐怕也不在少數。因此對於自己的孩子的立坐，行等姿勢也就不去注意，更談不到施以特殊的敎育與指導了。平常我們不是常常可以看見有些人的圓形的背，或者特別向

蔣介石以一個內地小商人的兒子，竟做到了中國的總司令和政治首領，這是很可耐人尋味的。他的殺戮極嚴厲可怕；但得他赦免而仍令服務的政敵卻也不在少數。他靠了日本的幫助

美娟在北里有悠久的歷史，十年前已享盛名，那時她的芳巢，在會樂里。觴政之盛，無出其右，與小雙珠陳美美聲同，爲當時新會樂裏的金字招牌。

某一個年頭，美娟收了房間，嫁過一個俠少。不過爲時不久，她又重歸舊調，美娟的舞術很佳，曾一度改行在維也納做舞女，

Fig. 2: Manually cropped text blocks

## The Corpus

Our corpus consists of 9.385 scanned folds from the entertainment newspaper Jing bao 晶報 (The Crystal), published 03.03.1919–23.05.1940 (Fig. 1). The double-keyed text ground truth comprises all April 1939 issues (40 folds, ~245.000 characters). Aside from text blocks and their headings, it also contains mastheads, advertisements and marginalia, however, the methods presented below will solely focus on “header-less” text blocks of uniform font-size.

## Character Segmentation

Due to Chinese characters’ nearly squared appearance, it is common to find resulting text blocks implicitly displaying a grid layout (see Fig. 2). Deviation from the grid usually appears when additional characters had to be squeezed into one column or because of inaccurate printing. In order for the method described below to work, we manually sort out any text blocks that don’t adhere to the grid layout and then extract the corresponding ground truth section for every crop.

After adaptive binarization (kernel size: 125 px) we calculate horizontal and vertical projection profiles, cf. Fan et al. (1998). To perform deskewing, we find an angle α with α ∈ [-2.0,-1.5,...,2.0] such that rotating the image by α maximizes

$$\sum_i^{w-1}(c_{i+1}-c_i)^2+\sum_j^{h-1}(l_{i+1}-l_i)^2$$

where w and h are the width and height of the image, ci is the number of black pixels in the *i*-th column (= the corresponding value of the vertical projection profile) and lj in the *j*-th line.

After deskewing, we cut the gray-scale, non-binarized original text block image into single character images along separators defined by the following heuristic:

(1) Use the valleys of the vertical projection profile to define separators between the columns.

(2) Use the valleys of the horizontal (global) projection profile to define separators between the lines.

(3) For every column, produce another (local) projection profile. If a local separator lies within 7px distance of a global separator defined by (2), discard the global separator and only use the local separator; else only use the global separator.

The positions of the valleys are obtained by scipy.signal.find_peaks using a minimum distance of (1) 22, (2) 20 and (3) 14.

For normalization and contrast enhancing the following method is used:

1. Globally (whole crop): Employ partial adaptive thresholding: Every pixel whose gray-scale value is larger (= brighter) than the average of a surrounding 7x7-kernel is set to 255 (white). Separately, every pixel whose value is greater than the median of the image (called threshold below) is assumed to be a background pixel and set to 255. Every other pixel keeps its gray-scale value. Choosing the median arises from the supposition that there are more background than content pixels.
2. Locally (after cropping rectangles containing one character each): Ignoring white pixels, linearly re-scale pixel values from [minval,threshold] to [0,255], where minval refers to the darkest pixel in the image. This allows even for very lightly printed characters to appear darker and have their decisive features more strongly separated from the background.

Finally, the resulting fields can be easily mapped to the ground truth text. Indentations have to be manually marked, and since the CNN (cf. Section 3.) requires squared images as input, we add white padding to transform the rectangular character images into square ones.

This method entirely relies on correct annotation. While we can easily detect errors like missing lines, this is harder for missing or extra characters within a line (checking the line length), and basically impossible for typos or swapped characters. To avoid such mistakes we can only double-check annotations, otherwise they lower recognition accuracy.

# Character Image Generation

The method described in the section above yields a total of 92.039 character images (47.986 train + 21.676 dev + 22.377 test). Due to the Zipfian distribution, we additionally present the following table:

Tab. 1

| x | number of characters with at least x samples |
|---|---|
| 1 | 3045 |
| 2 | 2355 |
| 3 | 1995 |
| … | … |
| 10 | 1091 |
| … | … |
| 20 | 696 |
| … | … |
| 50 | 301 |
| … | … |
| 100 | 137 |

Motivated by the low quantity of training samples for higher x, we generate additional synthetic training data and propose the following research hypothesis:

Pre-training on extensive amounts of suitably augmented character images will increase the OCR accuracy for evaluation on real-life character image data.

With the goal of imitating the real-life character images with artificial training data, we apply the following, partly randomized (in b., e2.2, f., g., and h.) augmentations to glyph images extracted from various fonts:

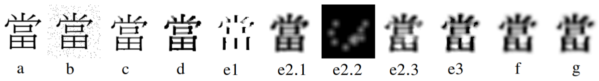


Fig. 4: Augmentations to glyph images

1. Extract PNG images of a predefined set of glyphs from a Song-Ti font (= the font-style used in the newspapers).
2. Add random noise (peppering).
3. Use morphological opening and then closing to enlarge noise pixels, grow them together with other close-by black pixels (other noise or the actual character) during erosion (= dilation of black contours on white background) and remove useless noise during dilation (= erosion of black pixels).
4. Use erosion to thicken lines.
5. Emphasize vertical lines while blurring and staining the remaining parts:
   1. Extract vertical elements of a certain minimum length using dilation with a vertical kernel.
   2. Separately apply the following:
      1. Further erode and blur the image.
      2. Generate random patches.
      3. Add the patches to the image.
   3. Join the result and the previously extracted vertical lines using bitwise AND.
6. Blur the image once more. Additionally, brightness can be randomly in-/decreased before. Afterwards, linearly rescale pixel values to cover the whole 0-255 range, like the real-life images.
7. Apply randomized elastic transformation.
8. Add padding and perform appropriate resizing.

Since ultimately, the classes used for OCR are Unicode points, the question arises which code points to synthesize additional training data from. We employ the simple heuristic of using all of the glyphs featured in the ground truth, and adding any missing ones from the 4000 most frequent characters of a representative corpus. Furthermore, inconsistencies caused by Han-unification have to be solved. For example, the image data features 靑 instead of 青 and 淸 instead of 清 (all different code points), however only one code point exists for every other character containing 靑/ 青 as a component (請, 情, 靜, …). While 值 and 値 (the latter being the variant used in our image data) have different code points, their right component itself (直) is Han-unified, etc. We decide to always use the most accurate code point as long as it’s not part of the CJK Compatibility Ideographs block (U+F900...U+FAFF), so e.g. 令 (U+4EE4) is used instead of 令 (U+F9A8), even though the latter might appear more accurate, depending on the font. Generally, we find that the character variants printed in our image data to be visually closer to the Japanese standard (e.g. the components 𠆢 and 示), so we choose several Japanese fonts for training data generation.

# Character Recognition

We decide on using a GoogleNet CNN architecture (Szegedy et al., 2015), slightly modified to take 1-channel inputs instead of RGB-images. This has proven to be effective regarding both printed and handwritten Chinese character recognition, e.g. Zhong et al. (2015) and Xu et al. (2018). Training on different character image sets, we obtain the following top-k accuracies on the real-life validation set for *k* ∈{1,...,10}:

Tab. 2

| k | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| only synthetic character images (4 different fonts) | 69.73 | 78.3 | 81.68 | 83.65 | 84.99 | 86.06 | 86.87 | 87.49 | 87.97 | 88.46 |
| only real character images | 96.47 | 97.29 | 97.46 | 97.56 | 97.61 | 97.66 | 97.68 | 97.69 | 97.69 | 97.71 |
| pretraining on synthetic; fine-tuning on real | 97.63 | 98.57 | 98.78 | 98.91 | 98.98 | 99.01 | 99.07 | 99.1 | 99.12 | 99.13 |

We also find that the selection of the fonts by which variants (i.e. mainland Chinese, Taiwanese, Japanese, Korean) it was designed for is largely negligible, i.e. a Taiwanese font may score higher than a Japanese font, even though the latter features glyph variants closer to those found in our data. This is probably because the percentage of characters with regional variants is relatively small, and also implies that the characters’ stroke length and distance as well as small variations in the size of single character components is more relevant to the OCR accuracy when evaluating on real-life character images.

Interestingly, while there is a huge difference in performance after training on synthetic vs. real data, the human eye is barely able to differentiate between even a big selection of synthetic and real character images if presented next to each other (cf. Fig. 5).

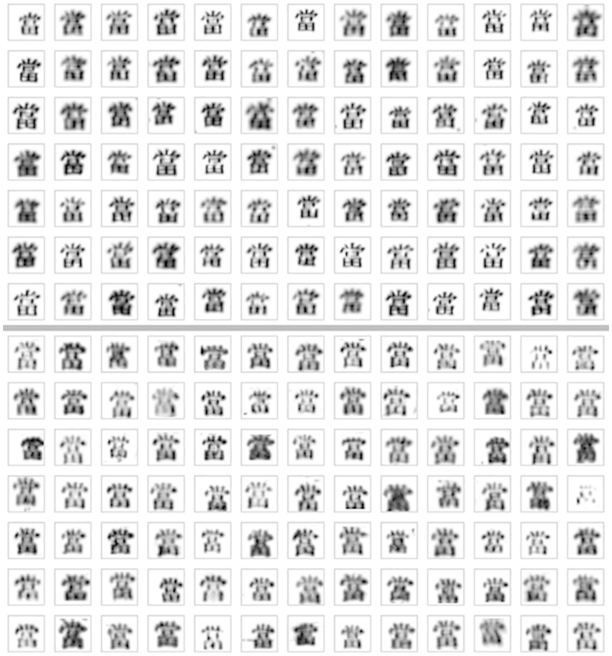


Fig. 5: Comparison between synthetic (top) and real (bottom) character images

# OCR Error Correction

Finally, we aim to improve top-1 accuracy values by using language models to find the correct character among the second to k-th prediction. As can be seen in the table in Section 3, there is a significant jump from top-1 to top-2 accuracy, meaning that for wrong predictions the gold character is often predicted in the second position.

Inspired by Wang et al. (2019), we propose a method that identi-fies characters likely to be incorrect: Let x1 and x2 denote the logit scores of the top 2 candidates output by the OCR model. Now we set a threshold t for the difference between x1 and x2. Any OCR prediction where x1– x2 < t is treated as likely to be incorrect and is passed on to the correction step. This step works by having a pre-trained BERT model re-predict the character from the top k OCR candidates. Systematically testing for different combinations of t and k (with t ∈ [0,0.5, . . . ,10] and k ∈ [0,1, . . . ,18]), we settle with t = 2.5 and k = 7, where we attain the following final results:

Tab. 3

| | Development set | Test set |
|---|---|---|
| Only OCR w/o pre-training | 96.54 | 95.49 |
| Only OCR w/ pre-training | 97.63 | 96.95 |
| OCR w/ pre-training + BERT-based correction | 98.05 | 97.44 |

As becomes evident, the presented post-processing method reduces the error by 18.1% (dev. set) / 16.1 % (test set).

# Bibliography

**Arnold, Matthias** (2021): *Ground Truth, Neural Networks, OCR: Towards Full Text of Republican China Newspapers*. https://tinyurl.com/ecpo-intro [letzter Zugriff 15. Juli 2021].

**Arnold, Matthias** (forthcoming): "Multilingual research projects: Challenges for making use of standards, authority files, and character recognition", in: *Digital Studies / Le champ numérique*.

**Arnold, Matthias / Hessel, Lena** (2020): "Transforming data silos into knowledge: Early Chinese Periodicals Online (ECPO)", in: Heuveline, Vincent / Gebhart, Fabian / Mohammadianbisheh, Nina (Hrsg.): *E-Science-Tage 2019: Data to Knowledge*. Heidelberg: heiBOOKS. S. 95–109. 10.11588/heibooks.598.c8420.

**Arnold, Matthias / Paterson, Duncan / Xie, Jia** (forthcoming): "Procedural Challenges: Machine Learning tasks for OCR of historical CJK newspapers", in: *International Journal of Digital Humanities.*

**Fan, Kuo-Chin / Wang, Liang-Shen / Tu, Yin-Tien** (1998): "Classification of Machine-Printed and Handwritten Texts Using Character Block Layout Variance", in: *Pattern Recognition* 31, S. 1275–1284.

**Holley, Rose** (2009): "How Good Can It Get? Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs", in: *D-Lib Magazine* 10.1045/march2009-holley.

**Liebl, Bernhard / Burghardt, Manuel** (2020): "From Historical Newspapers to Machine-Readable Data: The Origami OCR Pipeline", in: *Proceedings of the Workshop on Computational Humanities Research (CHR 2020).* Amsterdam, the Netherlands. S. 351–373. (= CEUR Workshop Proceedings). http://ceur-ws.org/Vol-2723/long20.pdf [letzter Zugriff 15. Juli 2021].

**Sung, Doris / Sun, Liying / Arnold, Matthias** (2014): "The Birth of a Database of Historical Periodicals: Chinese Women's Magazines in the Late Qing and Early Republican Period", in: *Tulsa Studies in Women's Literature* 33, S. 227–237.

**Szegedy, Christian et al**. (2015): "Going deeper with convolutions", in: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* Boston, MA. S. 1–9. 10.1109/CVPR.2015.7298594.

**Wang, Hsiang-An / Liu, Pin-Ting** (2019): "Towards a Higher Accuracy of Optical Character Recognition of Chinese Rare Books in Making Use of Text Model", in: *Proceedings of the 3rd International Conference on Digital Access to Textual Cultural Heritage.* New York, NY, USA: Association for Computing Machinery. S. 15–18. (= DATeCH2019). 10.1145/3322905.3322922.

**Xu, Xin et al.** (2018): "Chinese Characters Recognition from Screen-Rendered Images Using Inception Deep Learning Architecture", in: Zeng, Bing et al. (Hg.): *Advances in Multimedia Information Processing – PCM 2017.* Cham: Springer International Publishing. S. 722–732. (= Lecture Notes in Computer Science).

**Zhong, Zhuoyao / Jin, Lianwen / Xie, Zecheng** (2015): "High performance offline handwritten Chinese character recognition using GoogLeNet and directional feature maps", in: *2015 13th International Conference on Document Analysis and Recognition (ICDAR).* Tunis, Tunisia. S. 846–850. 10.1109/ICDAR.2015.7333881.