DHQ: Digital Humanities Quarterly
2019
Volume 13 Number 1

Modelling Medieval Hands: Practical OCR for Caroline Minuscule

Brandon W. Hawk <bhawk_at_ric_dot_edu>, Rhode Island College
Antonia Karaisl <antonia_at_rescribe_dot_xyz>, Rescribe Ltd
Nick White <nick_at_rescribe_dot_xyz>, Rescribe Ltd

Abstract

Over the past few decades, the ever-expanding media of the digital world, including digital humanities endeavors, have become more reliant on the results of Optical Character Recognition (OCR) software. Yet, unfortunately, medievalists have not had as much success with using OCR software on handwritten manuscripts as scholars using printed books as their sources. While some projects to ameliorate this situation have emerged in recent years, using software to create machine-readable results from medieval manuscripts is still in its infancy. This article presents the results of a series of successful experiments with open-source neural network OCR software on medieval manuscripts. Results over the course of these experiments yielded character and word accuracy rates over 90%, reaching 94% and 97% accuracy in some instances. Such results are not only viable for creating machine-readable texts but also pose new avenues for bringing together manuscript studies and digital humanities in ways previously unrealized. A closer examination of the experiments indicates regular patterns among the OCR results that could potentially allow for use cases beyond pure text recognition, such as for paleographic classifications of script types.

Introduction

In an age replete with digital media, much of the content we access is the result of Optical Character Recognition (OCR), the rendering of handwritten, typed, or printed text into machine-readable form. On a more specific scale, OCR has increasingly become part of scholarly inquiry in the humanities. For example, it is fundamental to Google Books, the Internet Archive, and HathiTrust, corpus creation for large-scale text analysis, and various aspects of digital humanities. As a number of recent studies and projects have demonstrated, the results of OCR offer a wide range of possibilities for accessing and analyzing texts in new ways.[1] 1

OCR has shifted the range and scope of humanistic study in ways that were not possible before the advent of computers. As stated by a team of scholars led by Mark Algee-Hewitt at the Stanford Literary Lab, “Of the novelties produced by digitization in the study of literature, the size of the archive is probably the most dramatic: ...now we can analyze thousands of [texts], tens of thousands, tomorrow hundreds of thousands” [Algee-Hewitt et al. 2016, 1] (cf. [Moretti 2017]). Beyond literary study specifically, new possibilities due to the mass digitization of archives have emerged across the humanities. Historians of books, texts, and visual arts (to name just a few areas) now have ready access to many more materials from archives than previous generations. Among the new pursuits of humanities scholars, computer-aided studies often no longer focus only on a handful of texts but encompass large-scale corpora — that is, collections of hundreds or thousands of texts.[2] Much of this is made possible by OCR. 2

Yet most studies and applications of OCR to date concern printed books (see, e.g., [Rydberg-Cox 2009]; [Strange et al. 2014]; and [Alpert-Abrams 2016]). The majority of text-mining projects in the humanities focus on eighteenth- and nineteenth-century printed texts. [3] One way to expand the potential for humanistic studies even further is to apply OCR tools to extract data from medieval manuscripts, but this area of research has received much less attention.[4] Indeed, the current situation with using OCR on medieval manuscripts is not much different from 1978, when John J. Nitti 3

claimed that “no OCR (Optical Character Recognition) device capable of reading and decoding thirteenth-century Gothic script was forthcoming” [Nitti 1978, 46]. Now, as then, there has been little progress on using OCR to decipher Gothic or any other medieval script, regardless of type, date, or origin.

This article presents the results of a series of experiments with open-source neural network OCR software on a total of 88 medieval manuscripts ranging from the ninth through thirteenth centuries.[5] Our scope in these experiments focused mainly on manuscripts written in Caroline minuscule, as well as a handful of test cases toward the end of our date range written in what may be called “Late Caroline” and “Early Gothic” scripts (termed “transitional” when taken together).[6] In the following, we discuss the possibilities and challenges of using OCR on medieval manuscripts, neural network technology and its use in OCR software, the process and results of our experiments, and how these results offer a baseline for future research. Our results show potential for contributing to not only text recognition as such but also other areas of bibliography like paleographical analysis. In all of this, we want to emphasize the use of open-source software and sharing of data for decentralized, large-scale OCR with manuscripts in order to open up new collaborative avenues for innovation in the digital humanities and medieval studies. 4

Medieval Manuscripts and OCR

The field of medieval studies, of course, relies on the transcription and editing of texts for analysis. Work with OCR on medieval manuscripts is potentially useful considering how many medieval texts remain untranscribed or unedited. In many cases, this is because of the unwieldiness of editorial projects dealing with hundreds of witnesses. In other cases, texts remain obscure because they are overlooked or ignored in the shadow of more canonical works. While digitization has expanded the size of the archive for humanities scholarship in substantial ways, medieval studies still has work to do before this archive may be used for examinations of understudied texts or large-scale analysis. 5

In this respect, we see the potential to harness open-source OCR software for medieval studies in a number of ways. Using OCR could help contribute to areas such as: the speed, efficiency, and accuracy of text transcription and editing; cross-manuscript comparisons of texts in multiple respects; paleographical analysis; studies of textual transmission; as well as corpus creation, searchability, and subsequent macroanalysis. For example, with the output of OCR from medieval manuscripts, we could compare versions of texts for editorial purposes with collation tools like JuxtaCommons, or for intertextual parallels within larger corpora with tools like Tesserae. With other methodologies in mind, using OCR for massive corpus creation would allow for text-mining with tools like Lexos or other statistical analysis software like R (as in [Jockers 2014]). Using OCR with medieval manuscripts also opens up new avenues of research related to scribal practices such as scripts and their variant spellings, irregular spacing, abbreviations, and errors — encompassing local and general considerations through diachronic and synchronic lenses. 6

Certainly there are difficulties to working with OCR software on medieval manuscripts that do not apply to printed books. Medieval scribes did not work with Unicode, nor even with modern typefaces. A few reasons why OCR has not been widely applied to medieval manuscripts, even though it is evidently a good idea, include irregular features of handwriting, abbreviations, and scribal idiosyncrasies; variations in spellings; non-standard page layouts; and deteriorated pages — among other issues that might appear. All of these factors make traditional OCR based on print technology difficult. Recently assessing the field of OCR for corpus creation in medieval studies, Michael Widner writes, “Medieval manuscripts are practically impervious to contemporary OCR” [Widner 2018, 132] (cf. [Hawk 2015]). However, with the availability of open-source Artificial Neural Network (ANN) technology, as well as an increasing amount of digitized sources available via open access, OCR for medieval manuscripts is actually becoming a greater possibility. 7

We believe that there is a significant point to using OCR software — rather than other tools for handwritten text recognition (HTR) — because of the relative regularity of medieval hands and scripts overall, which may be analogous in some ways to print typefaces in a general way.[7] Recent work has questioned the efficacy of OCR and turned to HTR as a preliminary way to analyze script types even before moving on to the process of recognition (e.g. by [Kestemont et al. 2017]). The differences between OCR (a long-established technology) and HTR (in an early phase of development) are subtle but substantial. The biggest difference lies with layout analysis and segmentation for processing: this stage is 8

usually built into the OCR engine, while it is a separate stage in HTR. OCR tends to be known for a focus on processing individual letter-forms or characters as isolated parts of the whole text or text lines and is more widely used on printed texts. HTR typically processes whole words or lines, and is upheld as the optimal technology for handwritten texts.

9 Yet OCR and HTR technologies and processes have recently converged to some degree. Major OCR engines are also used on handwritten texts in various ways. Indeed, there is now some amount of overlap between OCR and HTR in practice because of developments in machine learning; for example, engines like OCRopus allow for new methods and analysis and segmentation (see below, on the OCRopus engine). New modes of using OCR software with ANN technology shift beyond specific focus on characters. ANN technology has reached a point that makes this possible for several reasons, more than the previous type of technology. While both OCR and ANN technologies have been around for much longer than often acknowledged, we now have more computing power than ever before. The time seems right to use that technology for the challenges that medieval manuscripts pose to OCR.

10 At the outset, it is important to recognize that our goal is not to eliminate human analytical work with computers. But using open-source OCR tools for manuscripts has the potential to limit the time of editing and to increase the efficiency of dealing with large numbers of witnesses. As is obvious from other uses of OCR for text-mining, post-processing brings the need for cleaning up “dirty OCR” through means of what David Mimno has called “data carpentry” [Mimno 2014]. For print books, post-processing means, at the least, eliminating extraneous data such as chapter numbers, hyphens, page numbers, page headers and footers, and apparatus, as well as unwanted “noise” produced in the OCR process (see [Alpert-Abrams 2016]; and [Widner 2018, 132–4]). Manuscripts may include many of the same elements, as well as their own idiosyncrasies like glosses and marginalia. Surely the human is integral to this whole process.

Artificial Neural Networks for OCR

11 Artificial neural network (ANN) technology, also known as “deep learning,” “machine learning,” or just “AI” has been around for a surprisingly long time.[8] While recognized as a powerful method which could address a large range of problems in computing, its uptake was limited for years, in large part due to the need for very large training sets and general speed issues. However, its use has exploded in recent years as storage and computer processing is much cheaper, and in some sectors massive labelled training sets can be assembled automatically using the labor of unknowing web users (see, e.g., [Taigman et al. 2014]; and [Google]). ANNs are useful in a large variety of applications and are particularly good at dealing with fuzziness and uncertainty in data, identifying patterns in a noisy world.

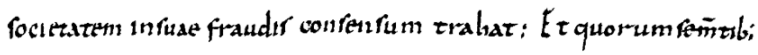
12 The basic idea behind ANNs is that they use algorithms to create a function that can perform some action, such as labelling different breeds of dogs, given a large training set, by repeatedly refining and testing different versions of a model. As this “training” process continues, the network becomes more and more accurate, until a point at which it plateaus and can no longer improve without more input data or changes in the initial configuration of the ANN. The model generated at each step is composed of many simple “yes/no” gates, automatically created by the training algorithms, which are given different weights through different iterations of the training process until an optimal configuration is found. These gates can be compared to biological neurons in the brain, which is where the name “neural network” comes from.

13 There are many types of ANNs, and each is appropriate for different uses. Deep neural networks (DNNs) use many hidden layers, which result in significantly more complex, but generally more accurate, recognition. Recurrent neural networks (RNNs) are a type of DNN in which different layers can refer back to each other, which has the effect of enabling some contextual “memory” — which is to say that later parts of recognition can make reference to earlier parts, and vice-versa. Finally, a type of RNN that has found much favor in OCR and many other applications is called Long Short Term Memory (LSTM). This is a clever configuration of a RNN such that contextual memory can endure for a long time when it is useful without skewing recognition results in other cases, which could happen with earlier RNN variants. The combination of a long contextual memory and a neural network makes LSTM very well suited to tasks like OCR and speech recognition, where the context (of earlier and later characteristics like pixels, sounds, characters, words, and beyond) is very helpful in inferring the correct result.

These ideas can be difficult to conceptualize in the abstract, so it helps to take a look at how they work in the case of OCR. The following description and accompanying images should help to make the process clearer. A training set will typically consist of many images, each containing a line of text, with accompanying “labels” that the ANN should somehow learn to associate with the parts of each image, until it can do this correctly even for unseen images. This can be done in different ways, so here we discuss how the open-source OCRopus engine works using a LSTM neural network. 14

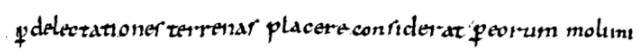
First, the page image used for training the neural networks is split up into the lines that make up the text (see Figure 1). Each line image is matched with a text transcription as a label — the UTF-8 formatted text corresponding to the text in the image, as transcribed by a human in order to train the computer how to recognize the text. All matching pairs of image and text lines taken together make up the so-called “ground truth” which is used to train the OCR model. This stage of the process creates image-text pairs, between the segmented line image and the ground truth transcription (labels), for the purpose of training the engine. 15

arrasbin/0010/010016.bin.png



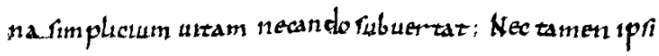
societatem in suae fraudis consensum trahat; Et quorum se m*tib;

arrasbin/0010/010017.bin.png



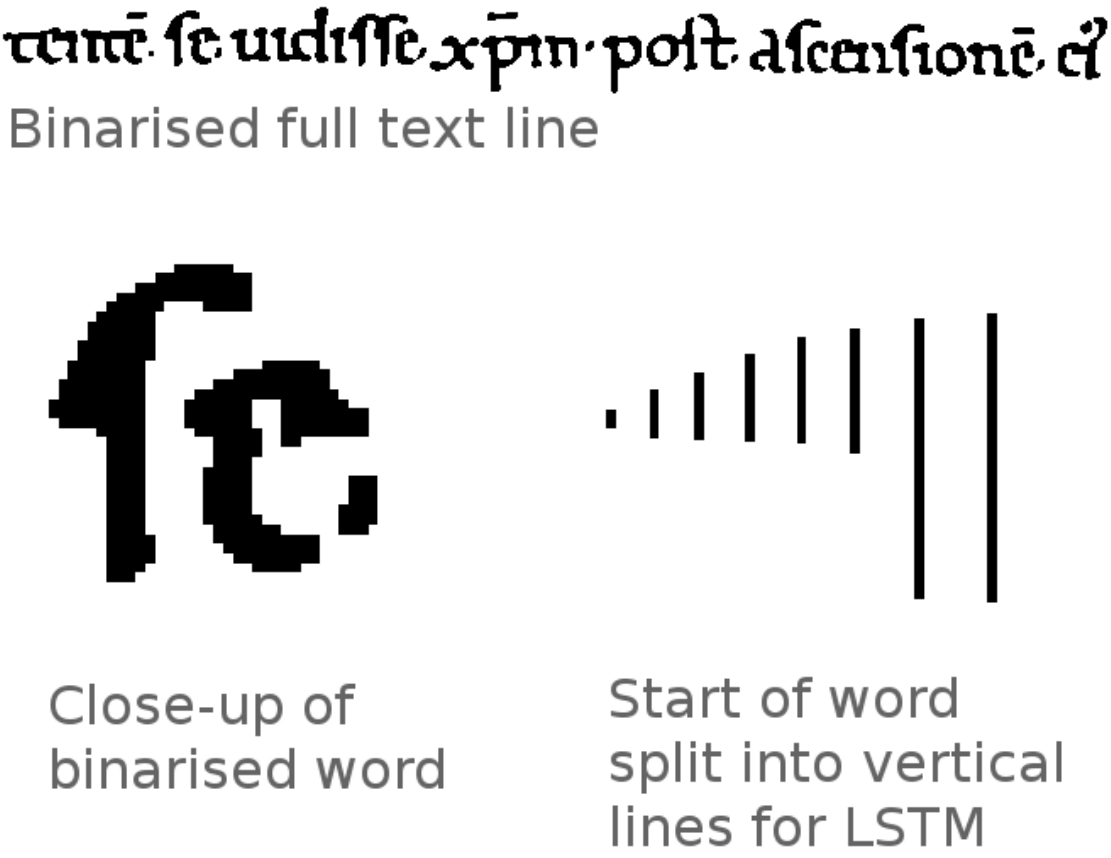
p delectationes terrenas placere considerat; p eorum molimi

arrasbin/0010/010019.bin.png



na simplicium uitam necando subuertat; Nec tamen ipsi

In the next step, the actual training process, each ground truth image line is split into vertical lines 1 pixel wide and fed into the LSTM network in order (see Figure 2). In this case the “memory” of the LSTM network refers to the vertical lines before and after a point being considered. The LSTM training engine will go through the lines one-by-one, creating a model of how each line relates to those around it, with the lines further away generally having less weight than those closest (but still able to influence things, thanks to the “long” memory aspect of LSTM). The exact configuration and weighting of different nodes of the LSTM network is altered many times during this process, each time comparing the result of OCR with this test network on the ground truth image lines. After a few tens of thousands of iterations, this process gradually finds the configuration which produces the best results for the whole corpus of ground truth. 16



By contrast, traditional OCR methods are based on rules defined by the designers of the system, defining algorithms for how best to split characters and match those characters to characters that have been previously seen. While intuitively we might expect human experts to do a better job at designing methods to do OCR than an RNN, recursive neural network systems consistently perform far better in practice. This is a well-recognized feature of recursive neural network systems [Karpathy 2015]. Again, perhaps surprisingly, neural networks are in particular very good at handling the fuzziness of real world inputs. Needless to say, this works in our favor for OCR in general, and for OCR of historical documents in particular. 17

While many medieval manuscripts are relatively regular in style and script, there is inevitably more variation in script than with printed documents. When coupled with the large number of variant spellings, irregular spacing, abbreviations, and errors used in the period, traditional OCR engines are not able to provide a good enough accuracy to be particularly useful. The power of OCR based on LSTM technology changes the situation, due to the significantly higher accuracy level in general, the ability to take as much context as is helpful into account, and the ability to better tolerate variation and other sorts of “noise” that crop up in centuries-old handwritten documents (see [Alpert-Abrams 2016]). 18

Of the RNN software currently available, our choice fell on OCRopus, an open-source program designed to train neural networks for optical character recognition. With respect to our purpose, OCRopus has some specific advantages vis-a-vis other open-source programs. First, it is distributed under the Apache License and freely usable by anyone under the outlined conditions. Second, OCRopus provides an easily modifiable set of commands that allows users to adapt the single steps of OCR (such as binarization, segmentation, training, and recognition) to their specific purposes. OCRopus is not a turn-key system, or software ready for use as is. Rather, this type of software requires the users to train their own models, the key input to the process; hence a degree of technical know-how is required. Even so, it offers the advantage that if any of the code has to be modified, or third-party programs used for discrete steps, this can be done in a simple, localized fashion without repercussions for the overall process.[9] 19

Unlike alternative OCR programs like Tesseract or Oculus, OCRopus’s neural networks are trained in a language-agnostic fashion — that is, exclusively focused on characters rather than words (see, e.g., [Baumann]; and [White 20

2012]). While OCRopus offers sets trained for specific languages like English, there is no corrective process interpreting strings of characters with the help of a set dictionary as a language-focused OCR program might. Although this sort of function can be of great use when adapting OCR for modern languages with a fixed orthography, this freedom from orthographic rules proves convenient when working with medieval manuscripts in Latin replete with variant spellings, irregular spacing, abbreviations, and errors. Indeed, for our purposes, this gives OCRopus an advantage over other OCR software and makes it a viable choice even in comparison with HTR technology.

Additionally, the documentation provided for OCRopus by both the original developers and users opens up the process significantly for further applications. One example is previous experimentation in using OCRopus with incunables by Uwe Springmann and Anke Lüdeling [Springmann and Lüdeling 2017]. Although incunables are printed, and of a greater regularity than expected from manuscripts, instructions and experiment results that Springmann and Lüdeling laid out provided an initial baseline for us to conduct experiments with.[10] 21

The open-source OCRopus engine uses a reasonably standard LSTM architecture. As mentioned above, it splits a page image into lines before passing the result to the LSTM engine, for either training or recognition. This is in contrast to most HTR systems, which cannot rely on lines being straight and mostly non-overlapping, and therefore have to rely on a more complex architecture. However, for printed texts and more palatable manuscript scripts like Caroline minuscule this is fortunately unnecessary, which means we can use a simpler architecture which is faster, smaller, and more accurate [Ul-Hasan and Breuel 2013]. The initial line-splitting is done with the tool “ocropus-gpageseg,”[11] which analyses a page image and outputs a series of lines in PNG format. One can then either perform recognition on these lines using an existing model, with the “ocropus-rpred” tool, or create a series of ground truth files that correspond to the lines (named, e.g., “imgname.gt.txt”) and train a new neural network model with the “ocropus-rtrain” tool. 22

Process and Results

1. Objective and theoretical approach

The basic objective of our experiments with OCR software based on neural network technology was to develop a workable, ideally time-saving, but sufficiently accurate solution for recognizing text in medieval manuscripts using open-source software. 23

At the start of our research, there were no known efforts publicly documented that would demonstrate the success of applying OCR to medieval manuscripts. Our initial experiments were therefore designed to demonstrate whether neural networks can be trained on medieval manuscripts in the first place, using ground truth created from target manuscripts. Results over the course of our experiments yielded character and word accuracy rates over 90%, reaching 94% and 97% accuracy in some instances. With the benefit of hindsight, these findings concur with the excellent results achieved with digitized manuscripts from the Royal Chancellory in Paris by the HIMANIS project, launched in 2017, albeit not with open-source software [Teklia 2017]. We might also compare these results to those of the Durham Priory Library OCR Project conducted using OCRopus [Rescribe]. 24

The viability of developing an open-source OCR solution based on neural network technology for medieval manuscripts can be considered proven in point. Even more, a model trained on the ground truth from a specific, single manuscript can yield high-accuracy results with that particular manuscript. We see this in the case of our experiment with Arras, Bibliothèque municipale 764, for which we achieved a shocking 97.06% accuracy rate for OCR based on ground truth transcription from this single manuscript. With a view to the primary objective of large-scale, collaborative work on manuscripts, the involvement of open-source software would be a desirable, expedient, and time-saving approach — ideally by developing a turn-key model. Rather than focusing on whether manuscript OCR is possible, our experiments tried to ascertain some best practices. 25

The hypothesis to be tested in our experiments, therefore, was that size and diversity of the training pool would be directly proportional to the quality of the resulting model when tested on “seen” and “unseen” manuscripts within the realm of Caroline minuscule scripts. “Seen” in the context of this article refers to test manuscripts included in the training 26

pool, albeit different pages; “unseen” refers to manuscripts not included in the training pool. While the results of seen manuscripts would be relevant in the research, developing a strategy for building a turn-key model for unseen manuscripts of a certain denomination is our ultimate goal. Among the specific types of application for neural networks, the strategic selection of training material seems to matter where there is no unlimited amount of data available, also depending on the type of material involved. That is, although the training pool can presumably never be big enough, the question remains about what kind of diversity is actually beneficial.

27 In the context of medieval manuscripts, the accuracy of the results is rarely satisfactory when a model trained on one specific manuscript is applied to an unseen test manuscript. This concurs with the result of Springmann and Lüdeling’s experiments with incunables [Springmann and Lüdeling 2017]. In a further step, however, Springmann and Lüdeling’s experiments sought to combine the ground truth from different incunables to observe the outcome. The experiment was designed with a diachronic corpus of texts spanning several centuries and the range of training and testing manuscripts was significantly broad. The mixed models developed yielded reliably worse results than the pure models where the OCR training pool contained ground truth from the target incunable only. This problem can be expected to only exacerbate in the case of medieval manuscripts where it is not only the different “font,” so to speak (the script), but also the idiosyncratic execution of the scribe (the hand) that sets one manuscript apart from another. Since single-manuscript models do not perform well for other manuscripts, however, the mixing of ground truth from different manuscripts would be the only way to develop a turn-key model for unseen manuscripts. The question is whether the trade-off between a decreased accuracy for seen manuscripts and time saved by using a turn-key model would be made worthwhile by sufficiently good results for unseen manuscripts.

28 Conceptually, it is useful to liken the situation of manuscript OCR to that of speech recognition technology. Handwritten texts are deliberately similar within script categories, but without a definite blueprint they all refer to; conversely, there is no such thing as the perfectly pronounced natural speech, only variations of it (male-female voice, old-young, accent-native). Within the pool of the spoken sound, there are boundaries defined by languages: within each, specific sounds (phonemes) will consistently map to specific letter combinations (graphemes). Across a range of languages, however, with different relationships of phonemes and graphemes, this consistent mapping would get confused. Accordingly, training pools for speech recognition algorithms tend to be limited to data in one language only.

29 Comparing the case of medieval manuscripts, not phonemes but characters (abbreviations included) map directly to letters or letter combinations from the Latin alphabet — at least to the extent that the relationship is consistent within each manuscript. Although medieval manuscripts can be written in a variety of languages, the linguistic boundary seems less relevant than the typographic one: across a range of hands, some characters of different scripts resembling each other correspond with different letters of the Latin alphabet. The Visigoth **a**, for example, looks more similar to a Caroline **u** than our expectation of an **a**. Defining the boundary of the training pool should therefore exclude instances where two characters from separate medieval hands resemble each other, but map to different letters (whereas mapping one Latin letter to two different-looking characters would not necessarily pose a problem).

30 Following that analogy, we limited the training pool mainly to one script type. Given its spread in Western Europe, across a range of geographies and time periods, Caroline minuscule naturally suggested itself as a more easily read and widespread script. Eventually, we did include manuscripts that pushed beyond initial models based on Caroline minuscule. In our second phase of testing, we also incorporated manuscripts containing scripts with features exhibiting later developments. Such manuscripts contain “transitional” Late Caroline and Early Gothic scripts (see the Appendix). These manuscripts helped to expand our experiments somewhat to begin seeing further results about the diversity of our training pool.

31 The implicit goal was to develop a model that could deliver a character accuracy of at least 90% on unseen manuscripts. As per our experience, certain technical parameters are crucial to achieve good output: good quality, high-resolution images; a minimum line height, even if that significantly slows down OCRopus’s processing in the segmentation and recognition steps; and eliminating skewed results from large illustrations, difficult layouts, and discolorations. The quality of both testing and training manuscripts can skew the interpretation: for example, a testing manuscript of extremely high quality might compare all too favorably and skew expectations with regard to the average

outcome. Within our overall framework, the goal was to establish tendencies rather than definite results.

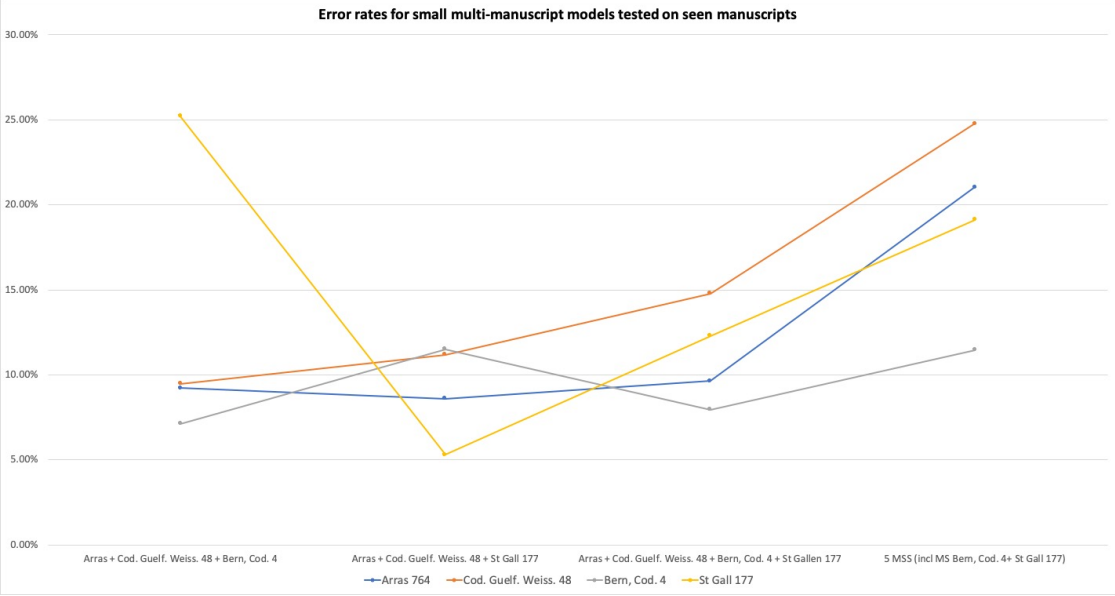
For the current project, the expansion of medieval abbreviations has been treated as a post-processing problem and bracketed out from the experimentation objectives. With a view to future research, however, we believe there to be scope for LSTM models to be trained to correctly expand ambiguous abbreviations. 32

The overall experiment proceeded in two steps: testing models from training pools as small as 2-5 manuscripts, and pools of 50 manuscripts upwards. 33

2. Size and diversity in small models

In the first round of experiments, we combined single pages of a small number of manuscripts written in Caroline minuscule in a training pool and tested them on different pages from these seen manuscripts. In the process, models were built from 2-5 different manuscripts. This step very much resembled the experiments with mixed models done by Springmann and Lüdeling on incunables and our results concurred with theirs [Springmann and Lüdeling 2017]. At the time, our assumption was that diversity within a training pool would be unequivocally beneficial in all cases. When tested on target manuscripts, however, these small, mixed models performed uniformly worse on seen manuscripts than pure models — that is, these results directly contradicted our assumption that a greater variety in a training pool would always lead to greater accuracy of the results. 34

Looking at the results more closely for both seen and unseen manuscripts brings more nuanced tendencies to the fore. Foremost, we encountered a phenomenon we termed “relative preponderance” in the context: the proportionally higher or lower representation of a manuscript or subgroup of manuscripts in the training pool and the subsequent effect on the accuracy of the respective test manuscript or subgroup. For example, when testing on seen manuscripts, i.e. manuscripts represented in the training pool, accuracy decreased with diversity of the training pool (Figure 3). In other words, the lesser the relative preponderance of the seen manuscript in the training pool, the lower the accuracy of the outcome. 35

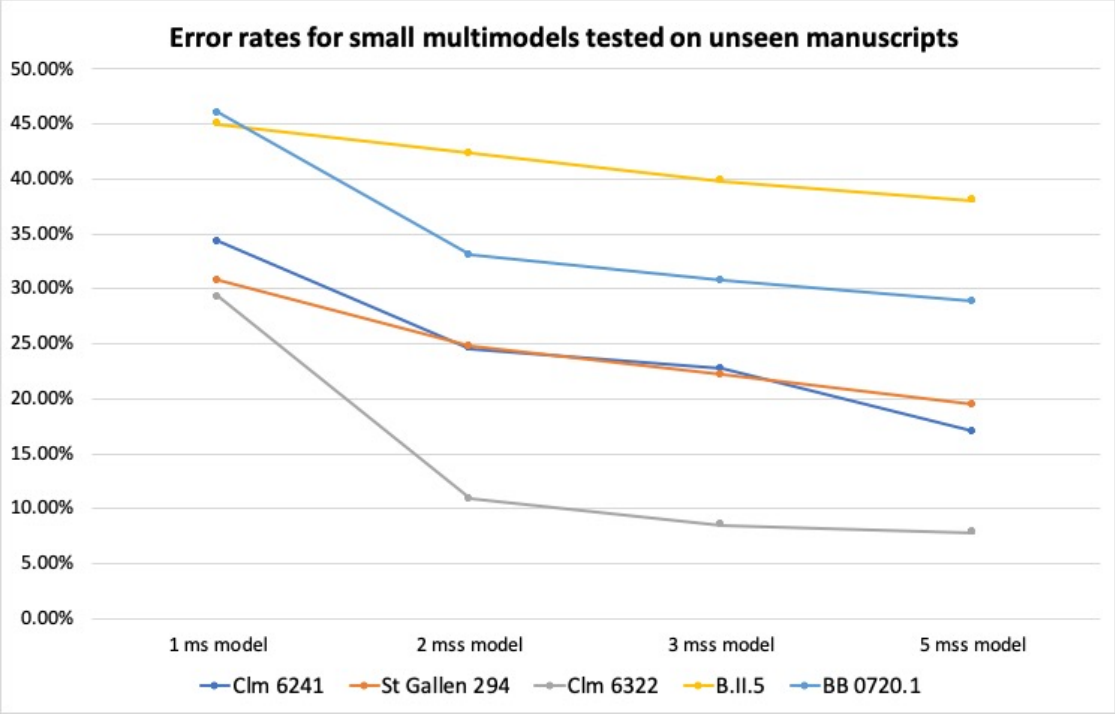


An exception to this rule was a model composed of 100 lines of Arras, Bibliothèque municipale 764 (c.800-1100, France and England), and around 100 lines of Wolfenbüttel, Herzog August Bibliothek, Weissenburg 48 (c.840-860, Weissenburg?). With an error rate of 2.33% when tested on unseen pages of Arras 764 the model outperformed the model built from 300 lines of Arras 764 combined with 100 lines of Weissenburg 48 (3.97% error rate) and the original model on only 300 lines from Arras 764 on the same test pages (2.53% error rate). The error rates in this case were surprisingly good for Arras 764 and, with further experiments, proven exceptional rather than a rule. A further elaboration on the experiment combined the original ground truth of Arras 764 with pages of ground truth from other manuscripts. The result shows that the model created from a combination of Arras 764 and Weissenburg 48 was some 36

kind of an outlier: in the majority of cases, the model formed from two different manuscripts did not outperform the original model trained exclusively on ground truth from the target manuscript.

A careful conclusion would have it that a variety of two or more manuscripts does not yield a better foundation for an OCR model for either target manuscript as a rule, but that specific combinations of manuscripts *can* yield exceptional results. In some cases, the reasons behind such results or the criteria for the respective manuscripts to be combined are not entirely clear. On the whole, mixing additional manuscripts into the training pool of a seen manuscript adversely affects the accuracy of the outcome. 37

When testing these models on unseen manuscripts, the trend was mostly the reverse: the more different manuscripts were included in the model, the better the results were for the unseen test manuscripts (Figure 4). As the graphed data shows, there is not a linear relationship between size and diversity of the training pool and the accuracy of the outcome. With so small a training and testing pool, the idiosyncrasies of each manuscript may also play a role. For the majority of the test cases, the results for unseen manuscripts were better for bigger training pools. 38



The general trend confirms that models tested on seen manuscripts yield worse results the more different manuscripts are added to the training pool. This finding goes against the grain of the assumption that a greater quantity and diversity of manuscripts within the training pool would make for a better model. Our explanation is that at such a low number of manuscripts combined, the accuracy in the results is chiefly determined by the relative preponderance of a seen manuscript in the training pool. In other words, the more its presence is diluted through the addition of more manuscripts, the lesser the accuracy in the end result. 39

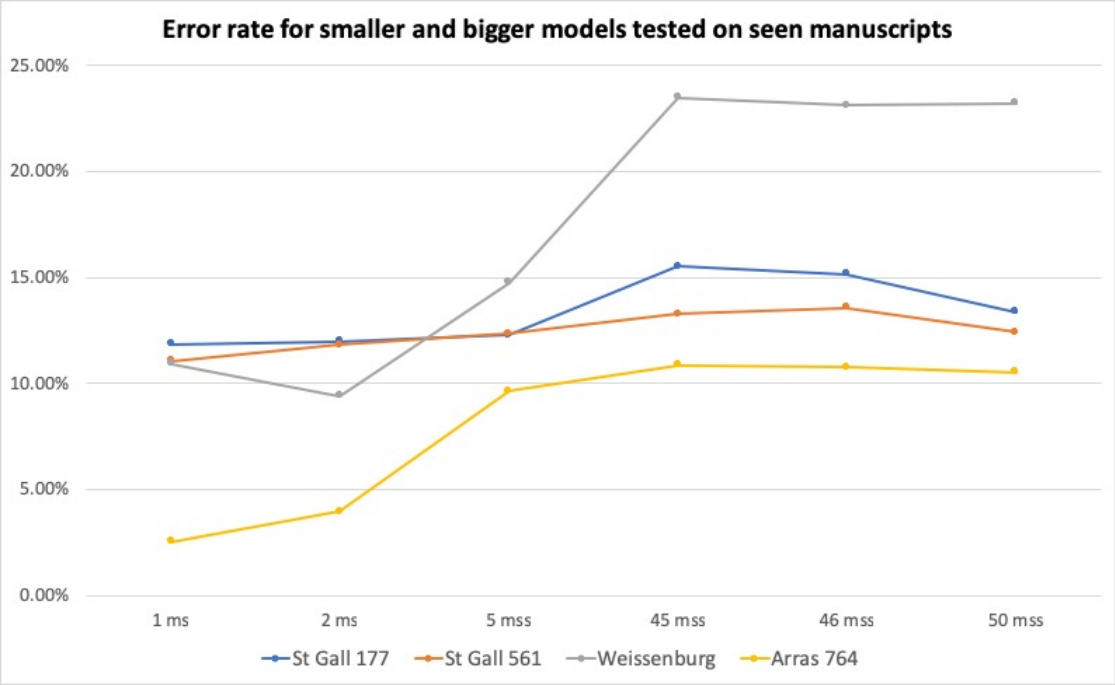
In the case of unseen manuscripts, the proportionality between quality of the model and diversity and size of the training pool holds. In this case, the relationship showed that the more different manuscripts were included in the training pool, the better were the results. Conceptually, our interpretation of this behavior is that while the relative preponderance of a seen manuscript is largely responsible for the accuracy for models of small size, the larger the diversity — and therefore complexity — in the training process, the better a model can deal with the unfamiliar hand of an unseen manuscript. Although the accuracy achieved with models of small size was not satisfactory, the findings suggest that the results would only improve with an increase in training material. 40

3. Size, diversity, and accuracy for bigger models

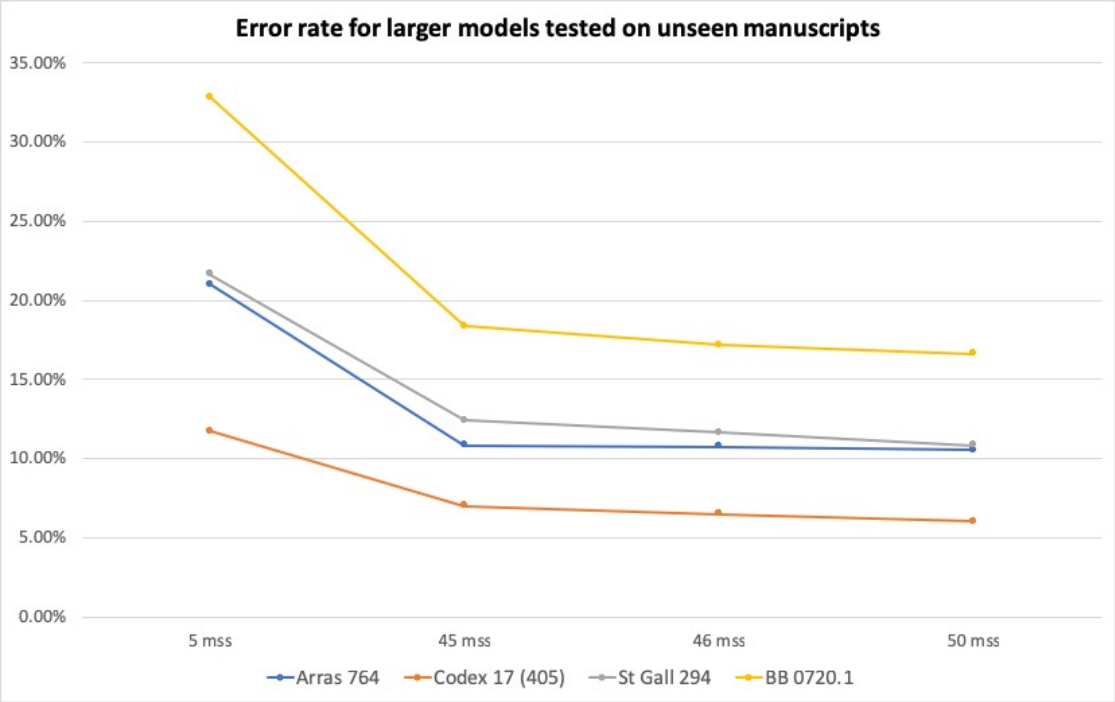
With the ultimate objective of our experiments in mind — building a turn-key OCR model applicable to as large a range 41

of unseen manuscripts as possible — the number of ground truth lines in the training pool were dramatically increased. The transcribed pages of 50 randomly chosen manuscripts of the time period between the ninth and eleventh century and written in Caroline minuscule or a transitional script were combined in one training pool and tested on seen manuscripts included in the training pool and unseen manuscripts not included in the training pool.

42 The results show that the relationship discovered between size of the training pool and the accuracy for the seen manuscript peaked (Figure 5). At some point, the relative preponderance of the target manuscript included in the training pool ceased to matter. Instead, the increase in size of the training pool was directly proportional to the accuracy achieved with the resulting model, almost uniformly.



43 The relationship between the size of the training pool and accuracy of the outcome continued to hold for unseen manuscripts (Figure 6). Yet looking at the improvement of the error rate, the experiments show that, in a small training pool, the addition of one or two manuscripts makes a veritable difference, while in the case of larger pools the difference made by each addition becomes marginal. In other words, while the improvement of model accuracy is proportional to the quantity of manuscripts included, this is not a linear development but a flattening curve.

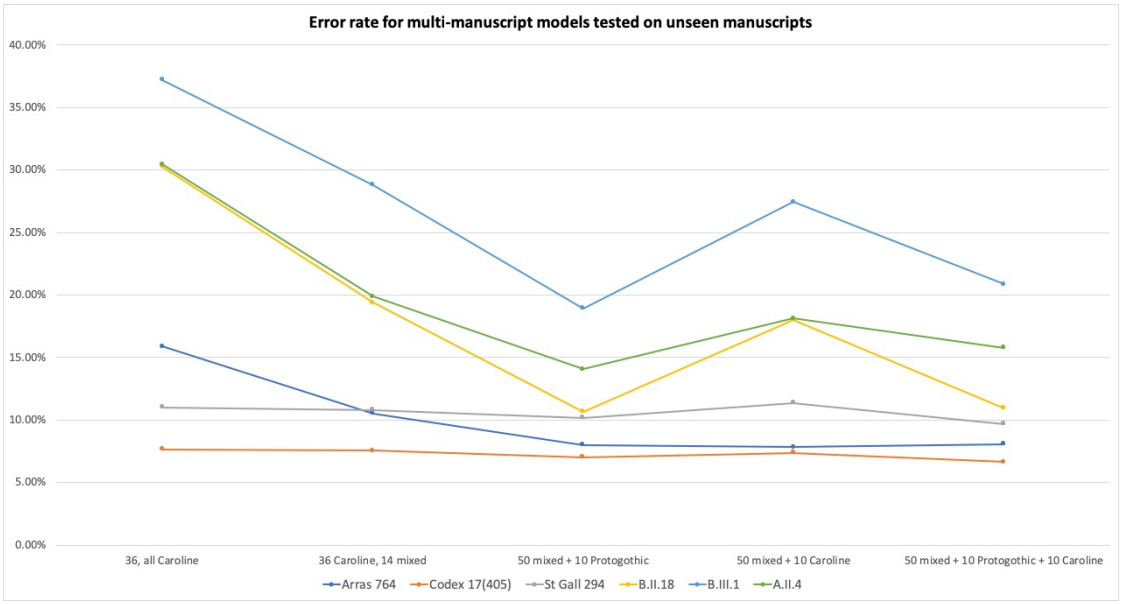


Originally, the training and testing pool had been limited to Caroline minuscule. However, other than with languages, the definition of Caroline minuscule is not set in stone: there are many manuscripts that show transitional stages manifest in different character forms, a growing number of abbreviations, and changing layouts. In a way, the boundaries between “pure” and “transitional” scripts are also not absolute but very much subject to human discretion — a circumstance that curiously comes to the fore when testing computational approaches to the area of manuscript studies (cf. [Kestemont et al. 2017]). 44

In a similar vein, a degree of variety was represented in our training set, with some of the manuscripts containing Late Caroline and Early Gothic scripts. A direct comparison of the training sets compiled from Caroline minuscule and strictly excluding transitional forms, versus the same training set including some transitional forms, brought to the fore the result that the presence of transitional forms in the training pool uniformly improved the output, rather than making it worse. Whether this improvement was simply due to the increase of size or diversity of the training pool could not be established for sure at this stage. 45

Conversely, the diversity of the training pool had a greater influence on the results with unseen manuscripts outside the immediate boundary of Caroline minuscule, in this case with transitional scripts. In other words, the relative preponderance of “outlier” manuscripts in the big training pool showed the same effect as the relative preponderance of the “seen” manuscripts in a small training pool. When tested on an unseen Early Gothic manuscript, the results for a lower number of manuscripts in the pool showed better results than with further, strictly Caroline minuscule manuscripts added. Similar to the idea of relative preponderance of the seen manuscript in the training pool, the proportional representation of “outlier” scripts in the training pool affects the results with unseen Late Caroline and Early Gothic test manuscripts. 46

This hypothesis was confirmed in further experiments (Figure 7). To the training pool of fifty predominantly Caroline minuscule manuscripts we added ten Late Caroline and Early Gothic manuscripts for one model, ten Caroline manuscripts for another, and one model combining all three groups. The results showed that for the predominant manuscript group in the pool (with Caroline minuscule script) increase in size and diversity of the model almost uniformly improved the results, more than the increase of size with a view to uniformity. In this set of experiments, our best results achieved an accuracy rate of 94.22%. 47



Differently said, in most cases, the addition of ten Late Caroline and Early Gothic manuscripts to the training pool (altogether combining a greater number of lines) made for better results than adding ten Caroline minuscule manuscripts. The difference of accuracy was very small in comparison but almost uniform across the Caroline minuscule test manuscripts. In all cases, the final, large model combining the original pool of fifty manuscripts, ten Late Caroline and Early Gothic, and ten Caroline minuscule performed best on all Caroline minuscule test manuscripts. 48

Conversely, the Late Caroline and Early Gothic test manuscripts uniformly performed best with the model combining the pool of fifty manuscripts with the ten additional Late Caroline and Early Gothic manuscripts, second best with the big combined model, and worst with the Caroline minuscule addition. Our conclusion is that the relative preponderance of Late Caroline and Early Gothic manuscripts in the training pool significantly influenced the outcome for “outlier” manuscripts, rather than the sheer increase in size of the training pool. 49

The resemblance in behavior with the example of seen manuscripts tested with small models suggests that there might be a similar peak whereby the relative preponderance of the outlier manuscripts in the training pool might cease to influence the outcome. Testing this hypothesis is not within the scope of this experiment but suggests an avenue for future research. 50

Lastly, it is notable that tests with “outlier” scripts other than Early Gothic — such as earlier manuscripts, for example — yielded neither remotely satisfactory results nor identifiable accuracy patterns across the test range of models. We found this to be the issue with two “limit case” examples from manuscripts written in Insular minuscule around the year 800. One example was a page from St. Gall, Stiftsbibliothek 761, a collection of medical texts, where none of our models achieved an error rate below 36%.[12] At this point in our research we have too little data to issue anything other than a hypothetical interpretation. Our conjecture is, however, that although overall stylistic distinctions between related scripts like Caroline minuscule and Insular (Caroline) minuscule are not considered immense according to paleographical accounts, even these smaller differences seem to have a big impact on the results with ANN OCR. The fact that these test cases failed to yield any substantial or regular OCR results suggests that the scope of the results of this experiment does not expand beyond the types of scripts used in the training data. These limit cases, then, demonstrate how a focus on script types is justified for further research; there is much more to be discovered in future experiments with this type of computer processing. 51

Given the regularity of behavior in the results with the Caroline and Early Gothic manuscripts, our experiments hint at how OCR technology based on neural networks might be used for further analytical purposes, with the potential to expand the same strategies beyond these script types. This became more apparent as we experimented with training and testing models made up of seen and unseen manuscripts, with a diversity of Caroline minuscule, Late Caroline, and Early Gothic scripts. Beyond pure text recognition (the main aim of using OCR software), results also indicate patterns 52

that could help to better understand paleography and categories of scripts. While our investigations so far have mainly focused on manuscripts with Caroline minuscule script, the outliers we incorporated define another set of results with broader implications than text recognition as such.

One particular example of wider applicability may be seen with results from incorporating manuscripts written in scripts from the eleventh, twelfth, and thirteenth centuries. After all, script types during this period are often described as "transitional," and paleographical distinctions between developments are difficult to classify (see [Derolez 2003, 56–71]; and [Kwakkel 2012]). The experiments with different types of training and testing models based on more or less script diversity and greater or lesser relative preponderance of manuscripts groups showed up regular patterns in different contexts that could be employed for analytical purposes. Using these patterns for paleographical analysis is a tantalizing prospect. We hope to pursue these possibilities in the future, as we build more training and testing models to bear out results from further diversity in our experiments. 53

Conclusions

Bibliography remains a fundamental basis of medieval studies, from traditional paleography to digital humanities databases. For example, research on scripts by paleographers like E. A. Lowe, Bernhard Bischoff, and R. A. B. Mynors formed the groundwork of the *Codices Latini Antiquiores* (*CLA*), which in turn gave rise to more recent endeavors in the digital age [Lowe 1934-1972] (cf. [Stansbury 2015-2018]). Without the *CLA*, we would not have examinations like Eltjo Buringh's attempts to statistically quantify and come to terms with the implications of the production and loss of medieval manuscripts across the centuries [Buringh 2011]. Similarly, without numerous bibliographical studies that came before, we would not have projects like *The Production and Use of English Manuscripts 1060 to 1220* by Orietta Da Rold, Takako Kato, Mary Swan, and Elaine Treharne, which provides valuable evidence about manuscripts, their contexts, and the history of books in early England [Da Rold et al. 2010]. 54

Using OCR software on medieval manuscripts both rests on and extends the work of bibliographers with new possibilities brought about by digital tools. In all of this, it is important to acknowledge the human elements of digital humanities. These aspects include the starting point of research questions asked from a fundamentally humanistic perspective as well as the labor involved in data curation and carpentry, manuscript transcription and processing, and interpreting results to consider the payoff and possibilities that they provide for future pursuit. After all, digital humanities approaches are only possible through the combination of computers with the types of critical, analytical research questions that drive the humanities. OCR extends the investigation of books while also connecting bibliography with other fields for greater possibilities of innovative pursuits. It is our hope that using OCR with medieval manuscripts offers a new range of questions built on bibliography but with fresh implications for future research using computers. 55

Using OCR with medieval manuscripts also helps to confirm and feed back into traditional bibliographical analysis in fields like paleography. One result of using digital software on medieval manuscripts is the confirmation that scripts as paleographers have described and identified them do appear and function distinctly (cf. [Kestemont et al. 2017]). The process of using OCRopus for our experiments has reinforced that certain groups of manuscripts with different script types (for example, Caroline minuscule, Late Caroline, and Early Gothic scripts) do differ from each other within general categories, not only in traditional paleographical assessments but also in the ways that the software handles them. OCRopus works differently in processing manuscripts exhibiting earlier Caroline minuscule features than it does with manuscripts exhibiting later Late Caroline and Early Gothic features, demonstrating that the software process is not the same in every case. In other words, how OCR software handles manuscripts differently within and across certain paleographical categories (even if they are relatively fluid) further justifies previous knowledge about such categories and their uses in medieval scribal practice. 56

It is clear that using OCR software with medieval manuscripts proves to be useful, but currently this is an area of bibliographical scholarship not well-served. Open-source software based on ANN technology can change this. Our experiments show that, given certain strategies, good OCR results can be achieved, even with a source pool of none-too-shocking size. We have initially experimented using mainly manuscripts written in Caroline minuscule, but our results from expanding to include others written in Late Caroline and Early Gothic indicate that these strategies could be 57

feasibly replicated across a larger corpus of manuscripts and shared alike with other processes. Some of the most exciting results from our experiments were in the best accuracy rates achieved with our corpus of manuscripts: 97.06% accuracy rate for results on the seen manuscript Arras 764 based on ground truth from only Arras 764; and 94.13% accuracy rate for results on unseen manuscripts based on ground truth from all training manuscripts. Our hope is that future work can build on these results to make OCR processing more efficient, more accurate, and more applicable to a wider range of medieval manuscripts.

58 The aim of future projects and collaboration using OCR with medieval manuscripts should be to share not only results but also data for the whole process so that others are able to build upon research for future improvements. While these are already fundamental aspects of many endeavors in digital humanities, the case of OCR for medieval sources underscores the necessity of open sharing. After all, our experiments and results are only one starting point, and our hope is that future research will improve the possibilities presented here. Furthermore, while we have used OCRopus for the experiments presented in this article, collaborative, open sharing allows for use with other OCR engines and different techniques. None of our results would be possible without open-source software, collaboration, open access to digitized manuscripts as facilitated by libraries, and, in some cases, the previous work of those experimenting with OCR. Like other ANN software, OCRopus does not come with training built in, so users need to train it. To build on OCR results, researchers need large training sets that have already been created and tested, so that they do not need to start from scratch every time. The future of this type of work necessarily calls for a collaborative approach to sharing data and results. We believe that this is where open-source software is more useful than proprietary software, which does not lend itself to building upon for collaborative work.

59 This sort of collaboration will require an amount of digital infrastructure and commitment to distribute training sets. Because training data for all of the popular ANN OCR software today is simply made up of pairings between source images (segmented lines) and ground truth text (transcriptions), they are easy to share. Sharing data in open access repositories is the most sensible approach, so that (for example) collaborators can access and use the data, identify and correct errors in training sets, and upload new batches of training data for different types of source analyses. In some ways, possibilities for this kind of sharing already exist in repositories that enable collaboration — like Github, Gitlab, and Zenodo.[13] This has also been our goal with sharing our own results in a Github repository. Our process using open-source OCR software with neural network technology should allow many people to participate collaboratively in decentralized fashion and on a much larger scale over time. Our intention with this process is to bring together different scholars, tools, and methodologies to build a robust, collaborative approach to OCR for medieval manuscripts.

Appendix: A Note on Manuscript Dates, Origins, and Scripts

60 For data about manuscript dates, origins, and scripts in our data sets, we have generally followed standard bibliographic descriptions in catalogues and online repositories. All of the data gleaned from paleographical analyses and descriptions are necessarily fluid (and often subjective). With this in mind, paleographical data should be understood as somewhat fuzzy: “meaning is under constant negotiation” [Kestemont et al. 2017, S108]. For example, all date ranges given are approximate, and any specific date given should be understood as “in or about” that year. Details that justify date ranges may be found in catalogues and other secondary literature about specific manuscripts. Our system for dating aligns with other bibliographic metadata standards, especially the “Principles of Description” in *The Production and Use of English Manuscripts 1060 to 1220* by Orietta Da Rold, Takako Kato, Mary Swan, and Elaine Treharne [Da Rold et al. 2010]. Dates for manuscripts in our data set are necessarily approximate.

61 The same need to account for fuzzy data is also true of script types, since the history and development of medieval handwriting is full of ambiguity. Classifications of scripts in our data are based on paleographical standards in bibliographic descriptions, but these are not always straightforward. Distinctions between script types, their characteristic features, and terminology are contested and rather difficult to pin down between about 1075 and 1225 (see details in [Derolez 2003, 56–71]; and [Kwakkel 2012]). Nonetheless, certain distinctive features do emerge during this period, and older forms drop out of use. Despite being slow and gradual, a transition does occur. For this reason, we take the view that Caroline minuscule was transitional in this period, and we use this term (following Kwakkel) throughout.

More specifically, we use two terms for scripts during this transitional period (both in this article and in our data sets): those exhibiting a balance of earlier features from Caroline minuscule, often before about 1200, are identified as Late Caroline; those exhibiting a balance of later features more like Gothic, often after about 1200, are identified as Early Gothic. Justification for these decisions is found in behavioral differences for how OCR software handles different types of scripts. These differences are most prominently seen in the way that adding different types of manuscripts (in Caroline minuscule, Late Caroline, and Early Gothic, as well as seen and unseen) affect the accuracy rates of OCR results (see our discussion in the section on “Process and Results”). While more detailed discussion of the implications of our results for paleographical analysis is beyond the scope of the present article, we hope to pursue these issues further in the future.

Notes

[1] There are many more than can be cited here, but see esp. results from the Stanford Literary Lab; [Moretti 2005]; [Moretti 2012]; [Jockers 2013]; and [Jockers and Underwood 2016].

[2] See previous references, esp. the overview and examples in [Jockers and Underwood 2016].

[3] For projects using OCR on early modern printed texts, see, e.g., [Rescribe]; and [eMOP].

[4] Some recent endeavors to render machine-readable data from medieval materials (manuscripts and incunables) stand out, although not all of these use OCR software: e.g., [Edwards et al. 2004]; [Boschetti et al. 2009]; [Fischer et al. 2009]; [Leydier et al. 2014]; [Hawk 2015]; [Springmann 2015]; [Springmann and Lüdeling 2017]; the Rescribe project [Rescribe]; [Camps 2017]; [Kestemont et al. 2017]; and the recently launched HIMANIS project [Teklia 2017], built on proprietary software by Teklia, not open-source tools that could be more widely applied.

[5] Data for our experiments may be found at https://github.com/rescribe/.

[6] On terminology and other issues, see our Appendix: A Note on Manuscript Dates, Origins, and Scripts.

[7] On distinctions between OCR and HTR, and critique of the former, see [Kestemont et al. 2017, S89-91]; and, e.g., [Transkribus], with documentation about both OCR and HTR at the project wiki, https://transkribus.eu/wiki/index.php/Main_Page.

[8] On ANN technology and its connection to OCR as discussed in this section, see esp. [Ul-Hasan and Breuel 2013]; [Simistira et al. 2015]; and [Ul-Hasan 2016].

[9] In our specific case, we chose to process the binarization with an alternative open-source program, ScanTailor, in order to maintain a higher resolution than OCRopus would produce. We also amended the numbering procedure for the segmented lines from hexadecimal to decimal as we realized that the initial hexadecimal order would confuse the order of lines in the raw text output.

[10] Our main sources for installation and training process, apart from OCRopus’s own documentation on Github, was Dan Vanderkam’s very useful contribution about installing OCRopus on Apple iOS [Vanderkam 2015] and Springmann’s detailed description of a training process for incunables [Springmann 2015].

[11] This tool, as are the others mentioned here, is built into OCRopus.

[12] See description and digital facsimile at *e-codices*, https://www.e-codices.unifr.ch/en/list/one/csg/0761

[13] See, e.g., the Github repository by [Kestemont]; and the Zenodo repository by [Springmann et al. 2018].

Works Cited

Algee-Hewitt et al. 2016 Algee-Hewitt, Mark, Sarah Allison, Marissa Gemma, Ryan Heuser, Franco Moretti, and Hanna Walser. “Canon/Archive. Large-scale Dynamics in the Literary Field.” Pamphlets of the Stanford Literary Lab, Pamphlet 11, January 2016.

Alpert-Abrams 2016 Alpert-Abrams, Hannah. “Machine Reading the Primeros Libros.” *Digital Humanities Quarterly* 10.4 (2016).

Baumann Baumann, Ryan. “Latin OCR for Tesseract.” https://ryanfb.github.io/latinocr/.

Boschetti et al. 2009 Boschetti, Frederico, Matteo Romanello, Alison Babeu, David Bamman, and Gregory Crane. “Improving OCR Accuracy for Classical Critical Editions.” In *Research and Advanced Technology for Digital Libraries*, ed. Maristella Agosti, et al., Lecture Notes in Computer Science 5714. Heidelberg: Springer, 2009: 156-67.

Buringh 2011 Buringh, Eltjo. *Medieval Manuscript Production in the Latin West: Explorations with a Global Database*. Global Economics History Series 6. Leiden: Brill, 2011.

Camps 2017 Camps, Jean-Baptiste. “Homemade manuscript OCR (1): OCRopy.” *Sacré Gr@@l: Histoire, philologie, programmation et statistiques*, February 6, 2017.

Da Rold et al. 2010 Da Rold, Orietta, Takako Kato, Mary Swan, and Elaine Treharne. *The Production and Use of English Manuscripts 1060 to 1220*. University of Leicester, 2010; last update 2013.

Derolez 2003 Derolez, Albert. *The Palaeography of Gothic Manuscript Books: From the Twelfth to the Early Sixteenth Century*. Cambridge Studies in Palaeography and Codicology 9. Cambridge: Cambridge University Press, 2003.

Edwards et al. 2004 Edwards, Jaety, Yee Whye Teh, David Forsyth, Roger Bock, Michael Maire, and Grace Vesom. “Making Latin Manuscripts Searchable using gHMM’s.” *Advances in Neural Information Processing Systems* 17 (2004): 385-392.

Fischer et al. 2009 Fischer, Andreas, Markus Wuthrich, Marcus Liwicki, Volkmar Frinken, Horst Bunke, Gabriel Viehhauser, and Michael Stolz. “Automatic Transcription of Handwritten Medieval Documents.” *Proceedings of the 2009 15th International Conference on Virtual Systems and Multimedia*. Washington, DC: IEEE COmputer Society, 2009: 137-42.

Google “Google reCaptcha.” 2016. https://www.google.com/recaptcha/intro/index.html.

Hawk 2015 Hawk, Brandon W. “OCR and Medieval Manuscripts: Establishing a Baseline.” *brandonwhawk.net*. April 20, 2015.

Jockers 2013 Jockers, Matthew L. *Macroanalysis: Digital Methods and Literary History*. Urbana, IL: University of Illinois Press, 2013.

Jockers 2014 Jockers, Matthew L. *Text Analysis with R for Students of Literature*. Switzerland: Springer International Publishing, 2014.

Jockers and Underwood 2016 Jockers, Matthew L., and Ted Underwood. “Text-Mining in the Humanities.” In *A New Companion to Digital Humanities*, ed. Susan Schreibman, Ray Siemens, John Unsworth. Malden, MA: Wiley-Blackwell, 2016: 291-306.

Karpathy 2015 Karpathy, Andrej. “The Unreasonable Effectiveness of Recurrent Neural Networks.” *Andrej Karpathy Blog*, May 21, 2015.

Kestemont Kestemont, Mike. “Code for the DeepScript Submission to ICFHR2016 Competition on the Classification of Medieval Handwritings in Latin Script.” https://github.com/mikekestemont/DeepScript.

Kestemont et al. 2017 Kestemont, Mike, Vincent Christlein, and Dominique Stutzmann. “Artificial Paleography: Computational Approaches to Identifying Script Types in Medieval Manuscripts.” *Speculum* 92/S1 (2017), S86-109.

Kwakkel 2012 Kwakkel, Erik. “Biting, Kissing and the Treatment of Feet: The Transitional Script of the Long Twelfth Century.” In *Turning Over a New Leaf: Change and Development in the Medieval Book*, ed. Erik Kwakkel, Rosamond McKitterick, and Rodney Thomson. Leiden: Leiden University Press, 2012: 79-125.

Lexos Lexos. Lexomics Research Group. Wheaton College. http://lexos.wheatoncollege.edu/upload.

Leydier et al. 2014 Leydier, Yann, Véronique Églin, Stéphane Brès, and Dominique Stutzmann. “Learning-Free Text-Image Alignment for Medieval Manuscripts.” In *Proceedings: 14th International Conference on Frontiers in Handwriting Recognition*. Los Alamitos, CA: IEEE Computer Society, 2014: 363-68.

Lowe 1934-1972 Lowe, E. A., ed. *Codices Latini Antiquiores: A Palaeographical Guide to Latin Manuscripts Prior to the Ninth Century*. 12 vols. Oxford: Clarendon Press, 1934-1972.

Mimno 2014 David Mimno. “Data carpentry is a skilled, hands-on craft which will form a major part of data science in the future.” *The Impact Blog*, September 1, 2014.

Moretti 2005 Moretti, Franco. *Graphs, Maps, Trees: Abstract Models for Literary History*. London: Verso, 2005.

Moretti 2012 Moretti, Franco. *Distant Reading*. London: Verso 2012.

Moretti 2017 Moretti, Franco. “Patterns and Interpretation.” Pamphlets of the Stanford Literary Lab, Pamphlet 15, September 2017.

Nitti 1978 Nitti, John J. “Computers and the Old Spanish Dictionary.” In *Medieval Studies and the Computer*, ed. Anne Gilmour-Bryson, special issue of *Computers and the Humanities* 12 (1978): 43-52.

Rescribe *Rescribe Ltd*. https://rescribe.xyz/.

Rydberg-Cox 2009 Rydberg-Cox, Jeffrey A. “Digitizing Latin Incunabula: Challenges, Methods, and Possibilities.” *Digital Humanities Quarterly* 3.1 (2009).

Simistira et al. 2015 Simistira, Fotini, Adnan Ul-Hassan, Vassilis Papavassiliou, Basilis Gatos, Vassilis Katsouros, and Marcus Liwicki. “Recognition of Historical Greek Polytonic Scripts Using LSTM Networks.” Presented at the *13th International Conference on Document Analysis and Recognition* (2015).

Springmann 2015 Springmann, Uwe. “Ocrocis: A high accuracy OCR method to convert early printings into digital text – a tutorial.” http://cistern.cis.lmu.de/ocrocis/tutorial.pdf.

Springmann and Lüdeling 2017 Springmann, Uwe, and Anke Lüdeling. “OCR of historical printings with an application to building diachronic corpora: A case study using the RIDGES herbal corpus.” *Digital Humanities Quarterly* 11.2 (2017).

Springmann et al. 2018 Springmann, Uwe, Christian Reul, Stefanie Dipper, and Johannes Baiter. “GT4HistOCR: Ground Truth for training OCR engines on historical documents in German Fraktur and Early Modern Latin.” August 12, 2018. https://zenodo.org/record/1344132.

Stansbury 2015-2018 Stansbury, Mark. *Earlier Latin Manuscripts*. NUI Galway. 2015-2018. https://elmss.nuigalway.ie/.

Strange et al. 2014 Strange, Carolyne, Daniel McNamara, Josh Wodak, and Ian Wood. “Mining for the Meanings of a Murder: The Impact of OCR Quality on the Use of Digitized Historical Newspapers.” *Digital Humanities Quarterly* 8.1 (2014).

Taigman et al. 2014 Taigman, Yaniv, Ming Yang, Marc’ Aurelio Ranzato, and Lior Wolf. “DeepFace: Closing the Gap to Human-Level Performance in Face Verification.” Presented at the *Conference on Computer Vision and Pattern Recognition* (2014).

Teklia 2017 *HIMANIS project*. Teklia. http://www.himanis.org/

Tesserae *Tesserae*. University at Buffalo. http://tesserae.caset.buffalo.edu/.

Transkribus *Transkribus*. READ project. https://transkribus.eu/Transkribus/.

Trettien 2013 Trettien, Whitney Anne. “A Deep History of Electronic Textuality: The Case of Eng/ish Reprints Jhon Milton Areopagitica.” *Digital Humanities Quarterly* 7.1 (2013).

Ul-Hasan 2016 Ul-Hasan, Adnan. *Generic Text Recognition using Long Short-Term Memory Networks*. Unpublished PhD dissertation, Technische Universität Kaiserslautern, 2016.

Ul-Hasan and Breuel 2013 Ul-Hasan, Adnan, Breuel, T.M. “Can we build language-independent OCR using LSTM networks?” In *Proceedings of the 4th International Workshop on Multilingual OCR*. Washington, DC: MOCR, 2013: article 9.

Vanderkam 2015 Vanderkam, Dan. “Extracting text from an image using Ocropus.” *danvk.org*. January 9, 2015. https://www.danvk.org/2015/01/09/extracting-text-from-an-image-using-ocropus.html.

White 2012 White, Nick. “Training Tesseract for Ancient Greek OCR.” *Eutypon* 28-29 (2012): 1-11.

Widner 2018 Widner, Michael. “Toward Text-Mining the Middle Ages: Digital Scriptoria and Networks of Labor.” In *The Routledge Research Companion to Digital Medieval Literature*, ed. Jennifer E. Boyle and Helen J. Burgess. New York: Routledge, 2018: 131-44.

eMOP *Early Modern OCR Project (eMOP)*. Texas A&M University. http://emop.tamu.edu/.