Tobias Englmeier, Florian Fink, Uwe Springmann and Klaus U. Schulz

Optimizing the Training of Models for Automated Post-Correction of Arbitrary OCR-ed Historical Texts

Abstract

Systems for post-correction of OCR-results for historical texts are based on statistical correction models obtained by supervised learning. For training, suitable collections of ground truth materials are needed. In this paper we investigate the dependency of the power of automated OCR post-correction on the form of ground truth data and other training settings used for the computation of a post-correction model. The post-correction system A-PoCoTo considered here is based on a profiler service that computes a statistical profile for an OCR-ed input text. We also look in detail at the influence of the profiler resources and other settings selected for training and evaluation. As a practical result of several fine-tuning steps, a general post-correction model is achieved where experiments for a large and heterogeneous collection of OCR-ed historical texts show a consistent improvement of base OCR accuracy. The results presented are meant to provide insights for libraries that want to apply OCR post-correction to a larger spectrum of distinct OCR-ed historical printings and ask for "representative" results.

1 Introduction

Most major libraries are currently engaged in *fulltext* digitization of historical printings. In this way, an important part of the world-wide cultural heritage will be made available in the internet for scholars and interested readers. Fulltext capture is based on optical character recognition (OCR). Unfortunately, OCR of historical printings has to face many problems, and still the overall quality of OCR-ed historical texts is unsatisfactory in many cases. If high standards are needed, often post-correction of OCR-results is inevitable (Nguyen, Jatowt, Coustaty, & Doucet, 2021; Dannélls & Persson, 2020; Magallon, Béchet, & Favre, 2018). Usually, the goal is to reconstruct the original spelling of each token in the printed document. Due to historical language variation and missing normalization of orthography, an *automated* OCR post-correction of this form turns out to be very difficult. When trying to improve accuracy a key problem is the avoidance of "infelicitous corrections" (misreplacement of correctly recognized OCR-tokens). Even if partial progress has been reached in international[1] and national projects[2] further work is needed to improve the situation.

[1] IMPACT Improving access to text, see http://www.impact-project.eu/

[2] OCR-D, Koordinierte Förderinitiative zur Weiterentwicklung von Verfahren der Optical Charakter Recognition (OCR), see https://ocr-d.de/

Modern technology for OCR and OCR post-correction is based on statistical approaches. The software for the two tasks is not static. Special models can be trained for a given application task, using a specific selection of ground truth materials for training. For OCR it has been shown that specialized models lead to much better accuracy results than models trained with standard ground truth corpora (Springmann, Fink, & Schulz, 2016; Reul, Wick, Springmann, & Puppe, 2017; Reul, Springmann, Wick, & Puppe, 2018). In this paper we investigate if for OCR post-correction a similar dependency of the quality of post-correction models on the form of ground truth materials used for model training can be observed. We concentrate on the kind of models employed in the post-correction system A-PoCoTo (Englmeier, Fink, & Schulz, 2019). In A-PoCoTo, a profiler service computes a ranked list of correction candidates for each unknown OCR-token. Employing a feature-based statistical model, the ranking is improved, before in a final step a second statistical model decides if the OCR-token is replaced by the best-ranked correction candidate or left unmodified (details s.b.).

This architecture, which is motivated by the avoidance of "infelicitous corrections", leads to distinct settings and feature systems available for profiling, re-ranking and decision step. Previous work on A-PoCoTo almost exclusively had been concentrated on the implementation of the system, only very limited initial evaluation results for one standard setting had been obtained. One goal of the present paper is fine-tuning and optimization. We want to explore in which way the selection of ground truth materials for training and all the other settings of the profiler and the training process influence the quality of the post-correction models obtained.

In order to understand our second goal, the typical situation of OCR-projects for historical printings in major libraries needs to be addressed. Often libraries wish to apply a single OCR and/or post-correction system to a huge collection of books and texts from distinct epochs. With respect to printing style, paper quality, filth, and language characteristics, printings show major differences. In order to judge the appropriateness of an OCR post-correction tool for real-life tasks, "representative" evaluation results for a very large spectrum of distinct OCR-ed printings would be needed. However, currently only a modest amount of ground truth data for older printings is available, and thus real "representativity" cannot be reached. As a step towards this ideal we selected a base collection of OCR-ed texts from distinct periods and with distinct OCR quality for fine-tuning and optimization. Afterwards, the evaluation is further extended to a larger class of documents to offer a more comprehensive picture.

We start with a brief introduction to A-PoCoTo and the features utilized for OCR post-correction in Section 2. Section 3 gives a more comprehensive view on the complete settings that determine the training of post-correction models for A-PoCoTo. In Section 4 we describe the corpora, ground truth data sets and evaluation scores for the following experiments. A classification of possible errors arising in automated OCR post-correction is added that helps to better analyze problems, weaknesses and difficulties. We also comment on some hidden inherent problems for each kind of evaluation.

Afterwards we first tackle the optimization task. For the experiments in Section 5

we use the ground truth data for the evaluation texts itself for model training. The results illuminate a hypothetical situation and limit where optimal ground truth data are available for training. Several variations studied address the full spectrum of alternative settings of the training process. In each case, the distribution of remaining post-correction errors is analyzed. Section 6 presents a parallel series of experiments where training and evaluation are based on disjoint parts of the same text. The models obtained can be considered as highly specialized.

In Section 7 we eventually train two general models derived from large ground truth data sets. The two models, which are not related to the evaluation texts, can be applied to any input. We compare accuracy results for these models with those obtained from optimized and special models. As it turns out, one of the two models ("19th century model") for all ten evaluation documents consistently leads to accuracy improvements comparable to those achieved with optimal ground truth.

In Section 8, following the second goal above, we extend the experimental basis. The 19th century model is applied to 20 documents from the pre-19th-century period. In a similar cross-validation experiment we apply general models similar to the 19th century model to 67 evaluation documents from distinct periods. In both series of experiments, ignoring two exceptions, we achieve improved accuracy for all documents.

After a review of related work in Section 9 we finish with a conclusion in Section 10.

2 The OCR post-correction system A-PoCoTo in a nutshell

A-PoCoTo is the automated component of the more complete post-correction system A-I-PoCoTo (Englmeier et al., 2019)[3] that combines an initial fully automated OCR post-correction step (A-PoCoTo) with an optional later interactive (I-PoCoTo) correction step. The interactive post-correction makes use of the insights obtained from the statistical analysis carried out in the automated correction step. The focus of the present paper are the statistical models and settings employed for the *automated* component A-PoCoTo.

2.1 Correction steps and statistical models of A-PoCoTo

In order to support detection and correction of OCR-errors, A-PoCoTo takes as *input* parallel OCR results obtained from two OCR-engines/models for a given historical text.[4] One of the OCR engines is destined as "master OCR", the other OCR acts as "slave OCR" that supports the master OCR.[5] Single tokens of the master and slave OCR will be respectively denoted in the form w_{mocr} and w_{slocr}. The following steps are shown in Figure 1, which gives an overview on both training and application/correction.

[3] A-PoCoTo is an open source tool under the MIT-license available at `https://github.com/cisocrgroup/ocrd_cis`.

[4] Working with two OCRopus-OCRs (Breuel, 2008) (one Master OCR and one supporting Slave OCR using different models) has at least two advantages: differences between the two OCR-outputs point to "suspicious" tokens. One of the two engines might provide the correct result.

[5] If initially just one OCR output is given and parallel images are available, A-PoCoTo may generate a second slave OCR output. Generalizations to the case of several slave engines are simple and are considered in other work.

Preliminary steps. As a preliminary step (both for training and application) the master OCR result and the slave OCR result for a text are aligned using the tokenization of the master OCR as a rigid base. In an unsupervised and fully automated way, a statistical profile is generated for the output of the master OCR using the **profiler** service developed in our group (Reffle & Ringlstetter, 2013; Fink, Schulz, & Springmann, 2017). The profile delivered provides for each non-lexical[6] token w_{mocr} of the master OCR of length at least 4 characters a ranked list of correction suggestions with scores. An expression $w_{cand,hist}$ stands for a single correction candidate for w_{mocr}. It should be mentioned that a correction suggestion may be identical to the OCR-token.[7] The correction candidates computed by the profiler and their scores are considered in the following steps. For efficiency reasons, only the ten best-ranked candidates are selected.

Automated correction. For the actual correction (application), the sequence of OCR-tokens of the master OCR serves as a starting point. Here we assume that two logistical regression models (**re-ranking model** and **decision model**, s.b.), obtained from training with a suitable parallel corpus containing OCR- and ground truth data, are available. Details are given below. Two steps are applied.

1. In the **re-ranking step** the selected profiler correction candidates for each token w_{mocr} of the master OCR of length ≥ 4 are ranked anew.[8] The **re-ranking model** trained before classifies *true* and *false* correction suggestions.[9]. For applying this model a set of features is generated for each pair of the form $\langle w_{mocr}, w_{cand,hist}\rangle$, also looking at w_{socr}. These features should yield an indication if $w_{cand,hist}$ is a plausible correction for w_{mocr}, given the parallel OCR result w_{socr}. Confidence values obtained for a correction suggestion $w_{cand,hist}$ range from -1 to 1. A value -1 indicates that a correction candidate almost certainly is a wrong correction suggestion, values close to 1 point to very plausible correction candidates. The new ranking positions of the correction suggestions $w_{cand,hist}$ for an OCR-token w_{mocr} are determined by the confidence scores obtained from the regression model.

2. The final step is the **decision step**. In this step, the definite decision is made whether an unknown token[10] w_{mocr} of the master OCR should be corrected using the best reranked correction candidate $w_{cand,hist}$ or not. The purpose of this step is to prevent A-PoCoTo from disimproving OCR results (avoidance of infelicitous corrections). If replacement is too eager, often correct OCR-tokens will be misreplaced. On the other hand, if replacement is too cautious, most incorrect OCR-tokens will be left unmodified. The **decision model** trained before is meant to distinguish between safe and hazardous replacements. Details of the features and the supervised training are described below.

[6] For tokens w_{mocr} that are found in the (modern or historical) lexicon integrated into the profiler, w_{mocr} itself serves as a single correction candidate. See (Reffle & Ringlstetter, 2013; Fink et al., 2017) for details.

[7] This may happen if the OCR-token is interpreted as a previously unknown historical variant of a lexical word.

[8] Recall that already the profiler comes with an initial ranking of correction suggestions.

[9] This model is obtained using *supervised* learning, using ground truth data, and comparing the two OCR outputs. The more rudimentary profiler ranking is based on *unsupervised* expectation-maximization, only analyzing the master OCR output.

[10] A token is unknown if it is not found in the historical lexicon of the profiler.

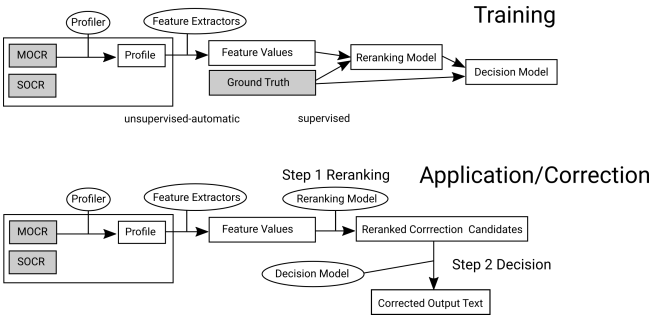

Figure 1: The full system A-PoCoTo. The visualization of training (upper part) is simplified in the sense that only one input text with parallel OCR outputs MOCR (master OCR), SOCR (slave OCR) and ground truth is shown. Both for training and application/post-correction, all correction suggestions are computed by the profiler. In the re-ranking step, the initial ranking of all correction suggestions generated for a particular master OCR token is improved.

As a general rule, OCR-tokens of length ≤ 3 are left unmodified. Even by editing a single symbol, a huge spectrum of alternative strings would be obtained, which makes correction difficult.

2.2 Standard feature set used for A-PoCoTo

For the two steps of the automatic post-correction in the initial setting considered before the start of this project we used the so called *standard feature sets*. These features are utilized both for training and for application/evaluation (cf. Figure 1). All standard features only operate on tokens w_{mocr} of length ≥ 4 (s.a.) of the master OCR that have at least one correction suggestion[11] and that are not dictionary entries. The features for the re-ranking step operate on the different correction suggestion of the profiler. For each token w_{mocr} of the master OCR the features examine each correction candidate $w_{cand,hist}$, also looking at the parallel token of the slave OCR w_{locr}, in order to determine the most likely candidate. In contrast, the features of the decision step operate on the most likely correction candidate as determined by the previous re-ranking step. For each token w_{mocr} of the master OCR these features examine the top-ranked correction suggestion in order to decide whether to correct w_{mocr} with the given suggestion or not. In what follows if a token w_{mocr} of the master OCR is given we write w_{slocr} for the parallel results of the slave OCR.

[11] there are tokens for which the profiler is not able to generate any correction suggestions; for these tokens no correction is attempted.

The full set of *standard features for the re-ranking step* is given in the Appendix. Here we only describe five features that turned out to be useful in an initial series of tests at the beginning of the project:

1 The *agreeing candidate feature* counts the number of tokens in the multiset $\{w_{mocr}, w_{slocr}\}$ that agree with the correction suggestion $w_{cand,hist}$.

2 The *OCR Levenshtein distance feature* yields the Levenshtein-distance[12] between w_{mocr} and w_{slocr}.

3 The *unigram candidate feature* yields the relative frequency of the correction suggestion in the master-OCR document.

4,5 The two Boolean *OCR match features* respectively indicate if w_{mocr} and w_{slocr} match $w_{cand,hist}$.

The *standard feature set for the decision step* is the same set as the one for re-ranking (cf. Appendix) with the following additional features:

6. The *ranking confidence difference feature* yields the difference between the re-ranker's confidence for the top-ranked candidate and the confidence for the second best candidate.

7. The *ranking confidence feature* yields the re-ranker's confidence of the top-ranked candidate.

2.3 Standard form of training for statistical models in A-PoCoTo

Before we can apply A-PoCoTo to an OCR-ed historical texts, two statistical models need to be trained.

1. Training of re-ranking step. For training a re-ranking model, a collection of historical texts is OCR-ed using a master OCR and a second slave OCR. In addition, a parallel series of ground truth versions for the texts serves as input. Each selected correction suggestion (computed for a token of the output of the master OCR by the profiler) is described in terms of the above feature system for re-ranking. Using the ground truth data we add the information if the correction suggestion is the correct token. Using this information a model is generated that is able to assign a confidence score to any selected correction suggestion described in terms of the same set of features. This score is meant to measure if a given correction candidate represents a plausible correction for a token of the master OCR.

2. Training of decision step. For training a decision model we assume that a re-ranking model is at our disposal and the correction candidates of each token of the master OCR (of length ≥ 4) are re-ranked. The best ranked correction candidate is

[12] The Levenshtein-distance between two strings u and v is the minimal number of symbol deletions, insertions or replacements that are needed to transform u into v.

described in terms of the above features for the decision step. We add the information if this correction suggestion is the correct token found in the ground truth.[13] The model computed in this way is able to compute a confidence score. This score indicates if it is reasonable to replace the OCR token by the correction suggestion.

Later, during post-correction, in our standard setting of the decision step we replace the OCR-token by the best-ranked correction suggestion if a score > 0.5 is achieved. Note that with this form of training, called *cautious training*, a correction candidate is classified as "wrong" even in cases where also the OCR-token does not correspond to the correct ground truth. In such a case it does not negatively affect accuracy when we replace the (wrong) OCR-token by the (wrong) correction suggestion. One may ask if the replacement strategy trained in this way is too cautious. In the experiments below we look at alternative forms of training for the decision step.

3 Settings influencing the training of post-correction models

Before starting the experiments on distinct training methods for the statistical models mentioned in Section 2.3 several preparations were made. Each of the following three points represents one degree of freedom for training improved statistical models.

3.1 Selection of feature set

We evaluated all features in the standard feature set (cf. Appendix) on the ten evaluation documents introduced in Section 4 below. We trained the re-ranking and the decision maker steps separately on one half of the according document and evaluated the overall performance of both (re-ranker and decision maker) trained classificators on the other half. In an initial series of experiments we compared the performance of the post-correction with different feature sets.

Afterwards we removed all features that either impaired the performance or appeared to not influence the overall performance of the classifiers. This left the following *base feature sets (bfs)* for the re-ranking and decision maker steps. Numbers refer to the above lists of features.

The **re-ranker base feature set** contains the *agreeing candidate feature* (1), the *OCR Levenshtein distance feature* (2), the *unigram candidate feature* (3), and the the *OCR match features* (4,5).

The **decision maker base feature set** contains the same features and in addition the *ranking confidence difference feature* (6).

New features

After the base feature set had been fixed, we took a look at some additional features for the classifiers.

[13] Recall that the best-ranked correction candidate can be identical to the given OCR-token. In this case it is of course irrelevant if we apply a replacement of not.

- The first additional features we evaluated are the two *Levenshtein distance features (ldf)*, which respectively give the Levenshtein distance between the OCR token w_{mocr} and w_{slocr} and the correction candidate.[14]

- The next additional feature we evaluated is the *correction position feature (cpf)*. This feature looks at the positions within w_{mocr} where error patters are applied in order to generate the given correction suggestion $w_{cand,hist}$. This feature measures the average OCR character confidence at these positions.

We also used the Calamari OCR engine (Wick, Reul, & Puppe, 2018) to generate token alternatives (with confidence scores) to each master OCR token w_{mocr} by using its voting mechanism. The resulting feature checks if $w_{cand,hist}$ is in this list of alternatives and uses the score in the positive case. Since the number of alternative candidates generated for each w_{mocr} is large, we experimented with different thresholds to narrow down the selection. However, for none of these settings the feature provided a significant advantage and so it was ultimately discarded.

In yet another series of experiments we utilized the word unigram frequency in a historical corpus as an additional feature for correction suggestions. No consistent improvement was observed.

In what follows, by an **extended feature set** we mean the base feature set enriched with the Levenshtein distance features, correction position feature, or both types of features.

3.2 Two versions of the profiler

The profiler in the form described above uses a background lexicon with modern and historical word forms collected before the start of this project. In our first experiments (s.b.) the analysis of post-correction errors showed that often the correct ground truth word was not in this lexicon and could not be generated as a correction candidate by the profiler. We thus prepared a new version of the profiler where we extended the background lexicon, using newly available ground truth data (Springmann, Reul, Dipper, & Baiter, 2018). As a matter of fact, ground truth data from the respective test documents were *not* used for this lexicon extension. In our experiments we studied the effect on post-correction when using the extended profiler lexicon both for training and evaluation. In addition, in the new profiler version we passed not only the ten best-ranked profiler correction candidates to the re-ranking step (s.a.), but all candidates.

In what follows, the **old profiler** refers to the profiler with the lexicon before the present project started. Only the ten best correction candidates are selected. The **new profiler** refers to the version with an extended background lexicon. All correction candidates are selected.

[14] Note that these features are different from the OCR Levenshtein distance feature in the base feature set. The latter measures the distance between w_{mocr} and w_{slocr}.

Year	Author	Titel	odd pt.	even pt.	WAC	CER
1487	Cuba	Garten der Gesunthait	2,478	2,591	0.57	0.34
1557	Bodenstein	Wie sich Meniglich	2,797	2,800	0.67	0.16
1588	Rosbach	Paradeißgärtlein	2,454	2,397	0.81	0.09
1652	Fabricius	Wund Artzney	2,333	2,363	0.69	0.18
1797	Wackenroder	Herzensergiessungen	25,315	25,259	0.18	0.62
1826	Eichendorff	Taugenichts	30,567	30,640	0.62	0.24
1841	Various	Grenzboten	20,316	20,553	0.82	0.06
1854	Keller	Heinrich I.	36,187	36,389	0.65	0.22
1870	Graßmann	Deutsche Pflanzennamen	4,335	4,383	0.81	0.11
1877	Saar	Novellen	31,909	32,195	0.41	0.36

Table 1: Documents, sizes (number of words), tokens accuracy WAC (percentage of correct OCR-tokens) and character error rate CER (percentage of misrecognized characters) of the test corpus.

3.3 Two training strategies

In the aforementioned standard form of training a decision model, we classify the best-ranked correction suggestion as "wrong" even if the corresponding ground truth token is also incorrect and a replacement would not hurt. Note that the correction suggestion could be even more similar to the ground truth than the OCR token. In what follows this form of training the decision model will be called **cautious training**. We introduced a second form of training (**courageous training**) where we ignored all training examples where both the OCR-token and the correction suggestion are wrong. In this way, for each positive (negative) training instance the active replacement of the OCR-token by the correction suggestion comes with a real improvement (disimprovement) of accuracy.

4 Evaluation data and principles

4.1 Corpora and ground truth data sets

For the evaluation we used the freely available GT4HistOCR corpus (Springmann et al., 2018). We OCR-ed the documents in the corpus using two different models. Then we selected ten documents from different time periods, ranging from the 15th to the 19th century. The documents are shown in Table 1. By purpose, documents were selected with a broad spectrum of OCR accuracies. While for most of the documents the recognition accuracy for words falls between 0.6 to 0.8, three documents have distinctly worse accuracies between 0.4 and 0.2 (see Table 1).

Division into training and test data

The ten documents were split into two halves (even part and odd part), where each half of a document contains about the same number of lines. To keep the results comparable all test/evaluation experiments were conducted on the odd parts of the documents. Table 1 shows the sizes (number of tokens) of all parts.

4.2 Evaluation scores

Our main evaluation parameters are OCR- and post-correction *word accuracy (WAC)*. In both cases, word accuracy gives the percentage of tokens in the OCR post-correction result that represent correct words of the ground truth.[15]

4.3 Classification of post-correction errors

Even when using the most sophisticated models for re-ranking and decision, automated post-correction in general does not lead to perfect texts. In order to compare the strengths and weaknesses of distinct models, a closer look at distinct types of errors made by post-correction is needed.

We first look at input tokens w_{mocr} that represent actual OCR-errors and analyse possible reasons that post-correction may fail to produce the correct token $w_{correct}$.

1. Short error. If the erroneous OCR token w_{mocr} has length ≤ 3, it is not touched by our post-correction system. In what follows we only consider erroneous OCR token w_{mocr} has length ≥ 4.

2. Profiler weakness. In some cases, the profiler (given the erroneous OCR token w_{mocr}) is not able to generate the correct word $w_{correct}$. There are two possible reasons.

(a) (*Missing correction candidate*). Weaknesses of the underlying modern and historical lexica and patterns employed for desribing historical language variation may lead to a situation where the profiler cannot generate $w_{correct}$. In addition, the profiler only tolerates a restricted number of OCR errors.

(b) (*False friends*). If the erroneous OCR token w_{mocr} accidently is among the lexical words known by the profiler the profiler does not generate alternatives.

3. Wrong candidate selection. Even if the correct token $w_{correct}$ is among all correction suggestions generated by the profiler, it may fail to be in the list of ten top-ranked correction candidates selected for re-ranking and decision (old profiler). Note that in the new profiler this form of error is excluded.

4. Wrong re-ranking. Even if $w_{correct}$ is in the list of selected correction suggestions for w_{mocr} it may fail to reach the topmost position after the re-ranking step.

5. Wrong decision. Even if $w_{correct}$ is the topmost correction candidate after the re-ranking step, the decision step can block the replacement of w_{mocr} by $w_{correct}$.

Another type of error may result if the OCR-result is correct ($w_{mocr} = w_{correct}$) and the profiler does not recognize that w_{mocr} is a correct (historical) word. The profiler may produce a list of correction suggestions, and re-ranking and decision steps may lead to a wrong replacement. This form of post-correction error will be called

6. Infelicitous correction. An implicit assumption in all the above cases is that an OCR-ed and post-corrected token corresponds to a unique token of the ground truth.

[15] In the presence of tokenization errors, a similar - but distinct - notion of word accuracy would be obtained when conversely measuring the percentage of GT words that correctly appear in the OCR post-correction result.

In practice, in OCR-ed historical texts tokens of the ground truth are often merged or split. Hence

7. *Word merges*,
8. *Word splits*

and other tokenization errors represent another possible source of problems.

4.4 Three hidden problems for evaluation

In the current context, obtaining "truly faithful" evaluation results is difficult due to three principal problems.

The character set problem. A tedious technical problem is caused by the distinct character sets used in electronic representations of historical texts. In general, OCR-results, ground truth, and profiler output for the OCR for a given text come from three distinct places, each using a specific set of historical characters. As a consequence, both in the OCR and in the post-correction result characters may be found that do not occur in the ground truth. In the evaluation, errors may occur that would dissappear when using an appropriate form of character mapping. However, distinct ground truth texts come with distinct character sets. Hence for each single case an individual character mapping would be needed. After checking various cases we found that in general only a small number of errors would be repaired when using such a mapping. Hence we decided to ignore the problem. Still it should be remarked that a slight improvement of accuracy would result when an appropriate mapping between character sets would be applied.

The ground truth problem. Since the documents considered in our context are not digitally born, ground truth needs to be prepared manually. Errors may occur. Not all ground truth data sets are perfect reconstructions of the text. As we have seen above, unusual characters may cause problems. The GT4HIST corpus selected for the experiments meets very high standards.

The alignment problem. For each evaluation, OCR-texts and post-corrected texts need to be automatically aligned with the ground truth data. However, often token borders found in the ground truth data do not correspond to token borders in the OCR-ed texts. In some cases, an "objective" alignment is not possible. Hence evaluation results are also affected by the alignment algorithm.

In our evaluation experiments we in fact met all these problems. Hence an absolutely faithful evaluation is a kind of illusion. Still the deviations caused by the three problems are minor. We did our best to achieve realistic and informative results.

5 Using optimal ground truth data

In the first series of experiments we want to study the principal limitations of post-correction models for A-PoCoTo. For the experiments described in this section we use exactly the same collection of OCR-ed texts and ground truth sets for training as for the tests. We also assume that the same kind of OCR-engines are used for training and

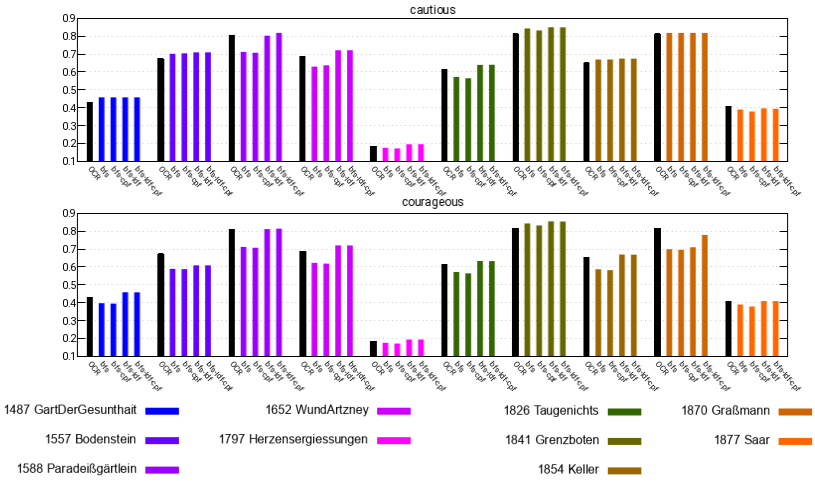

Figure 2: Word accuracy after post-correction when using *optimal ground truth, measuring the influence of the feature set and the form of training*. Each block represents accuracy results for one document obtained from OCR (black), post-correction with the base feature set (first coloured bar), and the extensions with the correction position feature cpf, Levenstein distance feature ldf, and both features, respectively (other coloured bars). The upper (lower) diagram refers to the *cautious* (*courageous*) correction strategy.

evaluation. This means that training and test data are identical, and the training data fit the application data in an optimal way. Of course in practice we hardly can have such a perfect correspondence. Hence we obtain a kind of upper border for the strengths of post-correction models. Even when using perfect ground truth data for training we cannot expect an optimal behaviour of the automated post-correction system: In fact, each post-correction model always represents a kind of *general* statistical rule that may lead to a wrong decision when applied to an *individual* OCR-ed token.[16]

5.1 Influence of feature set and training form on post-correction accuracy

Our first series of experiments illustrates the influence of (a) the choice of the feature set and (b) the form of training. In all cases, the *old profiler setting* (cf. Section 3.2) was selected.

[16] For example, a token may occur in an OCR-ed text two times in the same context, with the same confidence values, one occurrence being correct, the other one representing a recognition error. However, automated post-correction will treat both occurrences in exactly the same way. From the above description of A-PoCoTo we also see that lexical tokens are always left unmodified, which implies that "false friends" of the OCR are not detected.

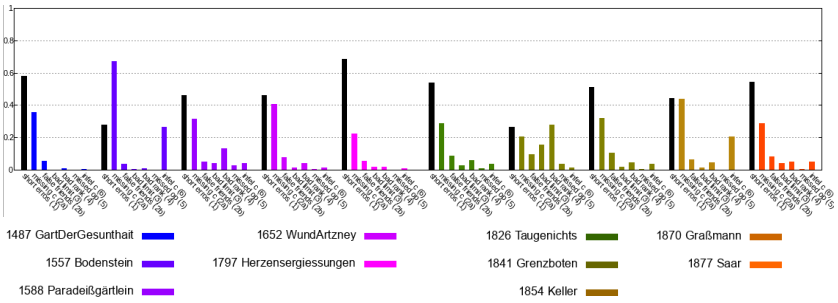

Figure 3: Distribution of post-correction errors when using *optimal ground truth*, courageous training, complete feature system. Most errors occur for tokens of length ≤ 3 (first bars) or are due to the fact that the correct token is not in the profiler lexicon (second bar).

Figure 2 shows the OCR word accuracy of our ten OCR-ed historical texts and the word accuracy reached after post-correction. For training and evaluation, in each case four feature configurations are utilized. For the first setting we only use the base collection (s.a.) of features (first coloured bar in each block). In the second and third setting we respectively add the correction position feature (s.a.) and the Levenshtein distance features (s.a.) to the base system. In the fourth setting, we use all features. Results are shown both for the cautious and the courageous form of training (cf. Section 3.3). The most important comments and insights are the following:

- Recall that for the post-correction tests a collection of documents with a wide variety of OCR accuracy values and dates of origin has been selected. Consequently OCR word accuracy is modest and, by purpose, shows large differences. For three documents, OCR word accuracy is below 50%. Three documents have an OCR word accuracy of (almost) 80% and better.

- When using the base feature collection for post-correction only, in some cases a clear disimprovement is resulting. The addition of new features leads to a clear improvement. For example, accuracy of document 1588 Paradeißgärtlein decreases from 0.81 to 0.71, approximately. When using more powerful feature collections, in almost all cases the OCR-word accuracy is in fact improved. Still, the improvement of OCR-ed texts is modest.

- For the cautious form of training (left diagrams) only one document (Saar) exists where accuracy could not be improved. For the courageous form of training, there are three such documents. For many documents the differences between the two forms of training are neglectable.

The results underpin the importance of the feature system employed for post-correction. For the ground truth and setting considered here, the cautious correction strategy seems to be preferable.

5.2 Distribution of post-correction errors

Figure 3 shows the distribution of post-correction errors for the above experiments when using optimal ground truth. We only consider the feature system with all features included. Since the error spectra respectively obtained for the cautious and courageous form of training are similar we only show results for the courageous form. Numbers are relative and represent percentages given the total number of all post-correction errors. Some important insights are obtained.

- *Short errors (1):* A very large part of all errors can be assigned to tokens of length ≤ 3. Note that our post-correction system does not address tokens of length ≤ 3.
- *Missing candidates (2a):* A substantial part of post-correction errors can be traced back to ground truth tokens that are not in the list of profiler suggestions. This shows that the weakness of the post-correction system to a large extend is caused by deficiencies of the language component of the profiler.
- *False friends (2b):* False friends cause a non-neglectable number of additional errors.
- *Bad limit (3):* Another part of the remaining errors is caused by situations where the correct string is in the list of all profiler suggestions but the rank is low and the string is not selected as the top candidate of the re-ranked list for the decision step.
- *Infelicitous corrections (6):* For most documents, the number of infelicitous corrections is small for both forms of training. Not surprisingly, the courageous form of training leads to a larger number of such errors, and when using this form of training some documents (Bodenstein, Graßmann) have a larger number of infelicitous corrections. This gives an explanation for the low accuracy results observed above for these documents (courageous training).

The results suggest to consider alternative variants of the profiler and motivate the "new profiler" setting described in Section 3.2 above.

For text representation, indexing and search in historical texts, tokens of length ≤ 3 in many cases are not interesting. We have seen that most of the erroneous tokens (for OCR and post-correction) have length ≤ 3. Figure 4 – as a pendant of Figure 2 – shows the picture where all accuracy values only refer to tokens of length at least 4. Naturally, under this focus improvements are larger. As to the distribution of errors types, now "missing correction candidate" is by far the dominating type, which again motivates the "new profiler" setting.

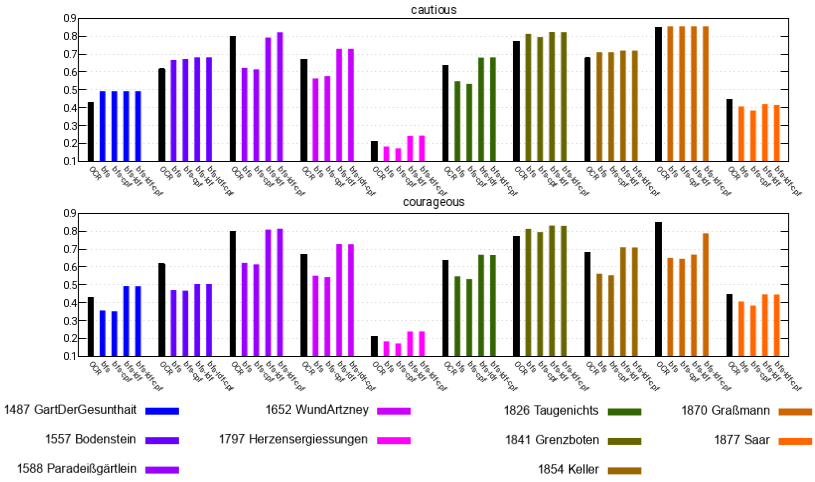

Figure 4: Word accuracy after post-correction for *tokens of length* ≥ 4 when using *optimal ground truth, measuring the influence of the feature set and the form of training*. The upper (lower) diagram refers to the *cautious* (*courageous*) form of training.

5.3 Influence of the profiler setting on post-correction accuracy

The above error analysis shows that the majority of post-correction errors is in fact independent of the choice of a post-correction model: ignoring problems with short tokens, the two error classes "missing correction candidate" and "bad limit errors" cover the majority of all errors, and other kind of feature systems and/or training data cannot help to get rid of these errors. In order to obtain more insights on how to influence the post-correction quality we built the new version of the profiler as described in Section 3.2.

Figure 5 compares the accuracy reached after post-correction when using the old and the new profiler setting. We again looked both at the cautious (upper diagrams) and the courageous (lower diagrams) form of training.

It is seen that the new version of the profiler leads to a clear improvement. Surprisingly, now the courageous form of training is slightly preferable! When using the full feature system, in fact now *all* OCR-texts are improved by the post-correction (bottom diagram). As a matter of fact, also for the new profiler setting the improvement of accuracy values is larger if tokens of length ≤ 3 are ignored. With the new profiler setting, when using the cautious form of training in some cases[17] the base set of features (first coloured bars in each group) leads to better results than the extended set of features. This surprising effect does not occur for the courageous form of training.

[17] Cf., e.g., 1797 and 1877 in the second diagram.

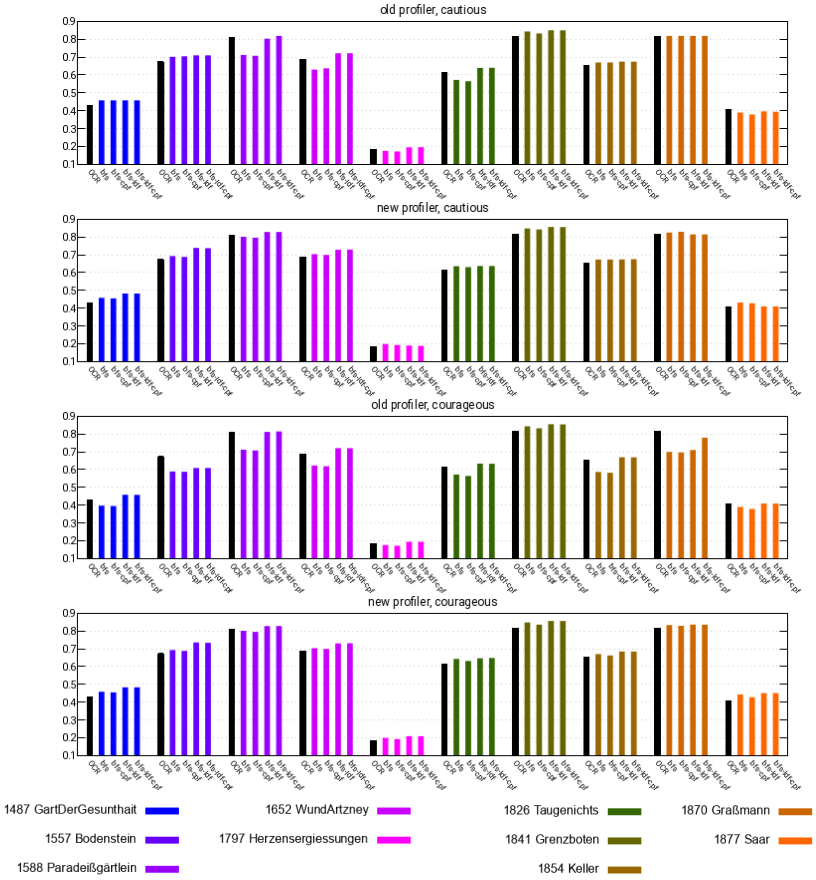

Figure 5: Comparing profiler settings: Word accuracy after post-correction when using optimal ground truth. Each block represents accuracy results for one document obtained from OCR (black), post-correction with the base feature set (first coloured bar), and the extensions with the correction position feature cpf, Levenstein distance feature ldf, and both features, respectively (other coloured bars). The first two diagrams refer to the cautious form of training, the last two diagrams refer to the courageous form of training. The second and the fourth diagram show the results for the new profiler.

6 Using specialized ground truth

In practice, when looking for a high quality specialized correction model for a specific OCR-text or corpus, a certain part of text can be processed/corrected manually, producing ground truth as a side result. The correction model derived from the OCR-result and the ground truth obtained can be used for automatically correcting the remaining part of the OCR output. Modeling such a scenario, in the following experiments ground truth data for training and evaluation are left disjoint, but always come from the same common source: the even part[18] of each document serves for training, and as in the above experients the odd parts serve for evaluation.[19] In this way we model a practical limitation for correction models. In all experiments we use the combination of the base feature set with the Levenshtein distance features and the correction position feature (cf. Section 3.1). As in the above case of optimal ground truth data we also look at the influence of the new profiler setting.

Figure 6 compares the accuracy reached after post-correction when using optimal ground truth with the accuracy reached with specialized ground truth.[20] In all triple groups, the black bar refers to the OCR accuracy, the first coloured bar shows the result for optimal ground truth, and the rightmost bar shows the accuracy for training with specialized ground truth. The two upper (lower) diagrams refer to the cautious (courageous) form of training. The second and the fourth diagram show the results for the new profiler.

When using the cautious form of training (upper part), the results for optimal and specialized ground truth are almost identical. For the courageous form of training, small differences become visible (e.g., first document, year 1487). In most cases, optimal ground truth leads to slightly better results than specialized ground truth. A possible explanation for the very small differences between the results for optimal and specialized ground truth is the close similarity between the odd and even parts of the underlying documents. Again the new profiler setting leads to larger accuracy improvements.

7 Using general ground truth corpora

For OCR and postcorrection, "general" models are based on larger amounts of ground truth data that are not related to the test documents. For our postcorrection experiments we trained two general models from two distinct corpora. The *pre-19th century model* was trained on documents prior to the 19th century and the *19th century model* was trained on documents from the 19th century.

For the training of the pre-19th century model we took all documents from both the *RIDGES-Fraktur* and the *Kallimachos* portion of the GT4HistOCR (Springmann et al., 2018) corpus that are dated prior to the 19th century and are not used in our evaluations.

[18] cf. Section 4.1

[19] Still we assume that for training and evaluation the same OCR-engines are employed.

[20] Optimal ground truth means that training and test data are identical, specialized ground truth means that training and test data rely on disjoint parts of the same document.

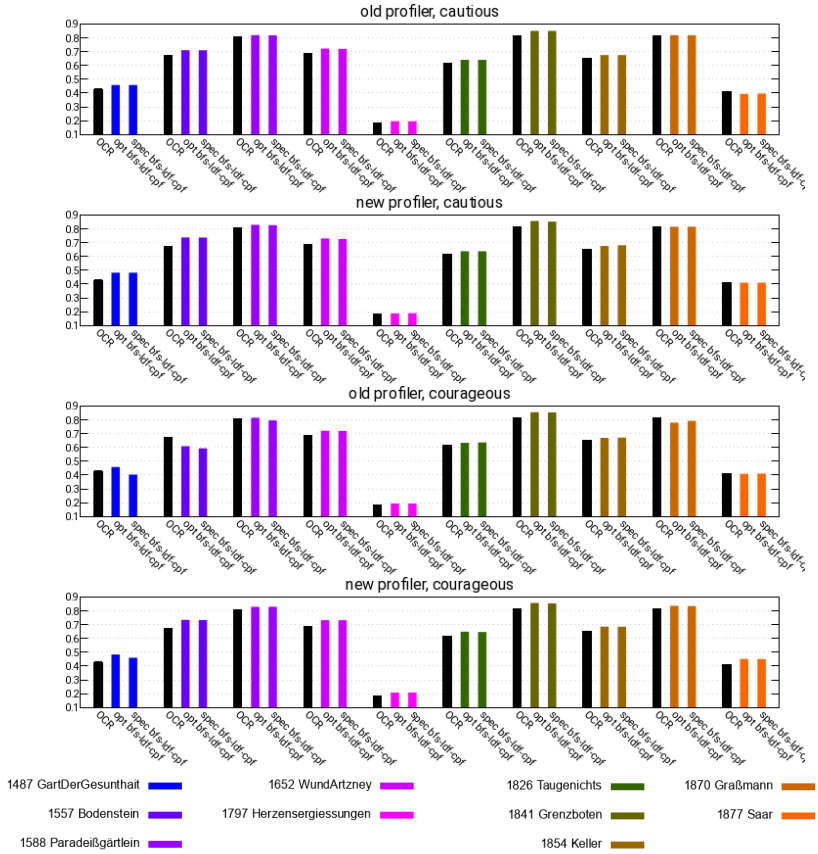

Figure 6: Comparing profiler settings: Word accuracy after post-correction when using *specialized ground truth* for training (complete feature system). Each block represents accuracy results for one document obtained from OCR (black), post-correction with the optimal ground-truth with the full feature set (first coloured bar), and post-correction with the specialized ground-truth with the full feature set (second coloured bar). The first two diagrams refer to the cautious form of training, the second two diagrams refer to the courageous form of training. The first and third line refer to the old profiler setting and the second and fourth line to the new profiler setting, respectively.

In the same manner we took for the training of the 19th century model all documents from both the *RIDGES-Fraktur* and the *dta19* portions of GT4HistOCR that are dated in the 19th century and are not used for the evaluation. In total the training corpus for the pre-19th century model contains 20 documents and the training corpus for the 19th century model 37 documents. The pre-19th century corpus has 179,267 tokens, the 19th century corpus has 1,881,093 tokens. Comparing these numbers with those given in Table 1 we see that the two general corpora, in particular the 19th century corpus, are much larger than the ground truth corpora utilized in the above experiments.

7.1 General models – word accuracy after post-correction

Figure 7 shows the word accuracy after post-correction when using general ground truth (second coloured bar in each block). Diagrams on the left-hand side refer to the pre-19th century model, diagrams on the right side to the 19th century model. Upper (lower) diagrams refer to the cautious (courageous) form of training.

Contrary to our expectation, there is no correlation between the time period of the ground truth corpus used for training and the post-correction quality for documents of a certain time period.

Generally speaking, best results are achieved when using the 19th century model and the courageous form of training (cf. bottom diagram on the right-hand side). In this setting, all OCR-documents are improved, and the improvements are almost identical to those reached with optimal models (first coloured bars).

We end up with an unexpected result. With appropriate settings, the power of general post-correction models can be absolutely comparable to those obtained when using "optimal" ground truth. Probably this effect is due to the large size of the training corpora utilized for the general models.

8 Towards a more representative picture - extending the experimental basis

We mentioned in the introduction that libraries often would like to apply a single post-correction model to huge collections of historical printings. From this perspective, results for just ten documents are not very informative. Following the second goal above, in this section evaluation results are extended.

8.1 Testing the 19th century model on other documents

As a first step towards are more representative picture we applied the 19th century model (courageous training, full feature set, new profiler setting) to the 20 documents that were selected to train the pre-19th century general model above. Figure 8 shows the results of the evaluation.[21] There are only two cases (texts 13 and 20) where the post-correction deteriorates a document's accuracy. In all other cases the post-correction

[21] The values for base OCR accuracy show large differences. This is not untypical for OCR on historical printings, where paper quality, paper transparency, special historical fonts, dirt, and other effects influence OCR quality.

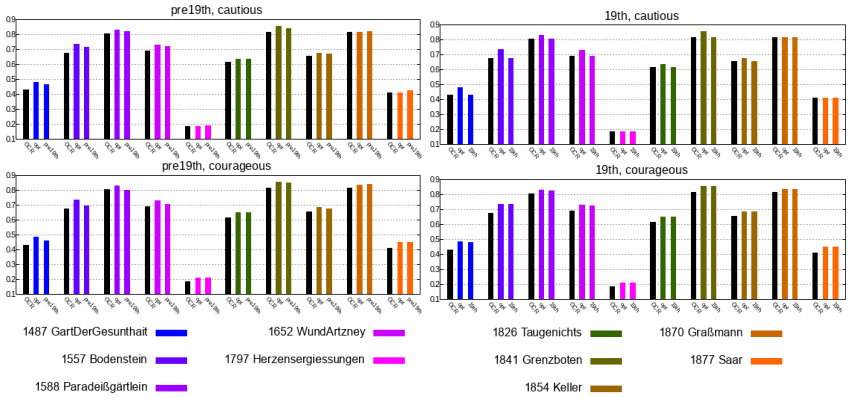

Figure 7: Post-correction with general models. Word accuracy after post-correction when using the *pre19th century model* (left-hand-side) and the *19th century model* (right-hand-side). In all cases, the complete feature system and the new profiler setting were selected. In each group, the accuracy reached with the general model is shown as second coloured bar, for the sake of comparison, the first coloured bar shows the parallel results when using optimal ground truth. The upper (lower) diagrams refer to the cautious (courageous) form of training.

improves the document's overall accuracy. In general it seems that the general 19th century model is fit enough to be utilized on a variety of different (pre-19th century) documents.

8.2 General models - cross-validation experiment for a large document collection

To further investigate the post-correction using general models, we cross-validated all the 37 documents from the 19th century corpus. For each single document we trained a general model using the remaining 36 other documents. The general models were trained using the courageous training setup, the complete feature system and the new profiler settings.

In this way we were able to test the power of general models on a larger number of documents. The models used by the cross-validation for the post-correction are similar enough to the 19th century model to draw conclusions about the general performance of such a model. Figure 9 clearly shows that post-correcting a document using a general model yields an overall improvement of the accuracy of each of the documents. Specifically there is no instance where the post-correction deteriorates the accuracy of a document, which implies that we are able to avoid too many infelicitious corrections.

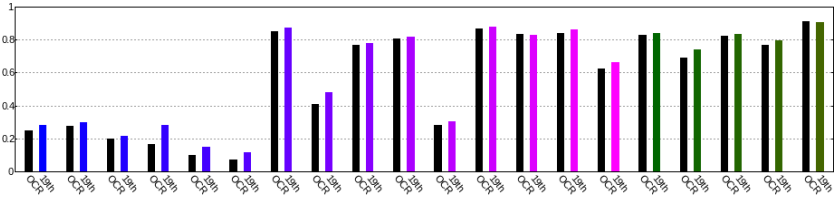

Figure 8: Evaluation on the 20 documents of the pre-19th-century corpus. Word accuracy after post-correction when using the *19th century model*. In each group the black bar shows the accuracy of the OCR before post-correction and the second (coloured) bar shows the accuracy reached with the 19th century general model.

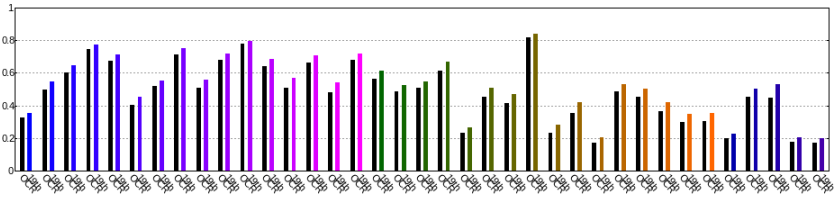

Figure 9: Cross-validation on the 37 documents of the 19th-century corpus. Word accuracy after post-correction when using the *cross-validation of the 19th century model*. In all cases, the complete feature system and the new profiler setting were used. The models where trained using the courageous training setup. In each group the black bar shows the accuracy of the OCR before post-correction and the second (coloured) bar shows the accuracy reached with the cross-validated general model.

9 Related Work

To get an overview of the many different approaches and techniques in the field of OCR post-correction, either the recently published paper by (Nguyen et al., 2021) or the older surveys from (Dengel et al., 1997) and (Kukich, 1992) are excellent choices. Also the two recent competitions on OCR post-correction (Chiron, Doucet, Coustaty, & Moreux, 2017; Rigaud, Doucet, Coustaty, & Moreux, 2019) conducted at the International Conference on Document Analysis and Recognition (ICDAR) give insights on the performance of different state of the art post-correction systems.

The data set used in the most recent 2019 competition (Rigaud et al., 2019) consists of historical documents of different domains and languages and its corresponding ground truth[22]. Our results described above cannot be compared directly with the competition results for two different reasons: First, in the ICDAR competition the post-correction task is strictly separated into the two steps of *detection of OCR errors* and *correction of OCR errors*. Second, the metric to measure the improvement of the correction step uses the weighted sum of the Levenshtein-distances between the correction and the ground truth, whereas we consider the total number of correct token before and after the correction procedure (WAC).

For our evaluation we chose a more holistic approach, in which we tried to measure the factual improvement of the post-correction on the documents. We are especially interested in the number of corrected words and not in the number of partially corrected words. Our evaluation puts special attention to possible infelicitous corrections that can harm the overall quality of the post-corrected documents. With the approach chosen in the ICDAR paper (separate evaluation of error detection and error correction) the total correction effect is not made transparent and it is not clear to which extend infelicitious corrections are avoided.

Only the raw OCR text and its alignment with the corresponding ground truth were made available to the competitors. The original images were not available, so that a system like A-PoCoTo, which is based on the input of several OCR engines, cannot be applied to that data set.

Nonetheless if image data would be provided it would be interesting to compare the performance of A-PoCoTo with the for instance context dependent system based on a pretrained BERT language model mentioned in (Rigaud et al., 2019). Other examples of context-dependent approaches built on neural networks are (Amrhein & Clematide, 2018; Magallon et al., 2018) or (Nguyen, Jatowt, Nguyen, Coustaty, & Doucet, 2020). These works were evaluated on either the ICDAR 2017 (Chiron et al., 2017) data set or the ICDAR 2019 (Rigaud et al., 2019) data set and were therefore trained to be applied to historical documents.

In addition several other post-correction systems which like our system rely on the inputs of multiple OCR engines exist. In these works different strategies of aligning and combining OCR inputs are explored: E.g. (Reul et al., 2018) make use of a voting

[22] The used data set also contains some documents from the GT4HistOCR corpus our system was trained on.

based approach to post-correct OCR-ed early printed books in which multiple models are trained and then a decision based on confidences values for recognized characters and their alternatives is made. (Wemhoener, Yalniz, & Manmatha, 2013) also rely on a voting based decision based on the alignment of multiple OCR outputs derived from copies of the same source documents, while (Al Azawi, Ul Hasan, Liwicki, & Breuel, 2014; Al Azawi, Liwicki, & Breuel, 2015) utilize LSTM networks to determine the final output tokens and evaluate their method on a data set consisting of more modern documents (Guyon, Haralick, Hull, & Phillips, 1997). Another approach which is based on the recognition of text reuse and the alignment of these repeated passages is that of (Xu & Smith, 2017) which was tested on a collection of 19th century newspapers.

10 Conclusion

In this paper we studied the training of models for post-correction of historical OCR-ed texts. Post-correction of OCR-ed historical texts tries to reconstruct the original historic spelling of words in the printed document. In view of the large variety of historic spelling variants there is a realistic danger that post-"correction" replaces correctly recognized OCR tokens by better-looking correction suggestions, even decreasing accuracy. All experiments were based on the open source post-correction system A-PoCoTo, the automated component of A-I-PoCoTo (Englmeier et al., 2019).

The first goal of our project was to fine-tune A-PoCoTo in such a way that post-correction consistently improves accuracy of OCR-ed printings with distinct characteristics. Following this line, experiments for ten OCR-ed documents from distinct periods with a wide spectrum of values for OCR accuracy were conducted. The detailed evaluation of errors of the automated post-correction component successively led to several improvements of the original system: we optimized the features system, introducing new features. The lexical basis of the profiler service utilized for A-PoCoTo was extended. The number of correction suggestions of the profiler for a non-lexical OCR-token that is selected for the re-ranking and decision step of A-PoCoTo was enlarged. Eventually, in later experiments the strategy used for training re-ranking and decision was changed from a "cautious" to a "courageous" form of training.

As a result of all these optimization steps a uniform correction model based on a large general training corpus of 19th century documents could be developed such that using this model for post-correction a significant improvement of the accuracy of all ten evaluation documents could be reached. In all cases the improvement of accuracy was similar to that achieved with smaller, optimal training data from the evaluation document itself.

Libraries, which sometimes have millions of historic printings, are interested in "representative" results. However, the amount of ground truth data available for post-correction experiments is very limited. As a partial step towards a future, more representative evaluation, extensions of the experiments with our general post-correction model showed that in a set with 67 test documents the accuracy of 65 documents could be improved. Admittedly, for these documents only a small increase of accuracy is obtained,

which implies that for achieving high accuracy a second *interactive* correction step would be needed. When using the full postcorrection tool A-I-PoCoTo the insights obtained from the automated step are used to simplify and speed-up interactive postcorrection.

We also studied if the training corpus of 19th century documents can be further improved for post-correcting individual documents by adding "optimal" ground truth data as those considered in Section 5. While for some texts a minimal progress could be reached, no consistent improvement was found. In other areas of Computational Linguistics, often general models (obtained from training with large data sets) can be further optimized for specific collections by using specialized training data. In our situation, each OCR-ed historic text comes with its own characteristics, e.g, in terms of spelling variants. Figure 7 (right lower diagram) shows that post-correction accuracy is almost identical when using (i) the general 19th century model and (ii) the model obtained from *optimal* ground truth data for the given text. Hence we assume that for short texts it is difficult to achieve any improvement of the general model. If for a *large* printed document suitable ground truth data are available it might be possible to improve a general model with a second training step. This is a possible point for future work.

The above comparison between the old and new profiler versions suggests that further improvements can be reached when using an *enlarged lexicon of historical spelling variants*. Also at this point, additional ground truth data would be needed. Missing ground truth data for historical texts with non-normalized orthography and the large amount of spelling variants in older texts also imply that at the present point powerful general contextual language models probably cannot be obtained. Recall again that the goal of the OCR post-correction is to reconstruct the original spelling in the printed document.

References

Al Azawi, M., Liwicki, M., & Breuel, T. M. (2015). Combination of multiple aligned recognition outputs using WFST and LSTM. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)* (pp. 31–35).

Al Azawi, M., Ul Hasan, A., Liwicki, M., & Breuel, T. M. (2014). Character-level alignment using WFST and LSTM for post-processing in multi-script recognition systems–A comparative study. In *International Conference Image Analysis and Recognition* (pp. 379–386).

Amrhein, C., & Clematide, S. (2018). Supervised OCR error detection and correction using statistical and neural machine translation methods. *Journal for Language Technology and Computational Linguistics (JLCL)*, *33*(1), 49–76.

Breuel, T. M. (2008). The OCRopus open source OCR system. In *Document recognition and retrieval XV* (Vol. 6815, pp. 120–134).

Chiron, G., Doucet, A., Coustaty, M., & Moreux, J.-P. (2017). ICDAR2017 competition on post-OCR text correction. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)* (Vol. 1, pp. 1423–1428).

Dannélls, D., & Persson, S. (2020). Supervised OCR post-correction of historical Swedish texts: what role does the OCR system play? In *DHN* (pp. 24–37).

Dengel, A., Hoch, R., Hönes, F., Jäger, T., Malburg, M., & Weigel, A. (1997). Techniques for improving OCR results. In *Handbook of Character Recognition and Document Image Analysis* (pp. 227–258). World Scientific.

Englmeier, T., Fink, F., & Schulz, K. U. (2019). AI-PoCoTo: Combining automated and interactive OCR postcorrection. In *Proceedings of the 3rd International Conference on Digital Access to Textual Cultural Heritage* (pp. 19–24).

Fink, F., Schulz, K. U., & Springmann, U. (2017). Profiling of OCR'ed historical texts revisited. In *Proceedings of the 2nd International Conference on Digital Access to Textual Cultural Heritage* (pp. 61–66).

Guyon, I., Haralick, R. M., Hull, J. J., & Phillips, I. T. (1997). Data sets for OCR and document image understanding research. In *Handbook of Character Recognition and Document Image Analysis* (pp. 779–799). World Scientific.

Hämäläinen, M., & Hengchen, S. (2019). From the paft to the fiiture: a fully automatic NMT and word embeddings method for OCR post-correction. *arXiv preprint arXiv:1910.05535*.

Kukich, K. (1992). Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, *24*(4), 377–439.

Magallon, T., Béchet, F., & Favre, B. (2018). Détection d'erreurs dans des transcriptions OCR de documents historiques par réseaux de neurones récurrents multi-niveau. In *25e conférence sur le Traitement Automatique des Langues Naturelles (TALN)*.

Nguyen, T. T. H., Jatowt, A., Coustaty, M., & Doucet, A. (2021). Survey of Post-OCR Processing Approaches. *ACM Computing Surveys (CSUR)*, *54*(6), 1–37.

Nguyen, T. T. H., Jatowt, A., Nguyen, N.-V., Coustaty, M., & Doucet, A. (2020). Neural machine translation with BERT for post-OCR error detection and correction. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020* (pp. 333–336).

Reffle, U., & Ringlstetter, C. (2013). Unsupervised profiling of OCRed historical documents. *Pattern Recognition*, *46*(5), 1346–1357. Retrieved from `http://www.sciencedirect.com/science/article/pii/S0031320312004323` doi: http://dx.doi.org/10.1016/j.patcog.2012.10.002

Reul, C., Springmann, U., Wick, C., & Puppe, F. (2018). Improving OCR accuracy on early printed books by combining pretraining, voting, and active learning. *Journal for Language Technology and Computational Linguistics*, *33*(1), 3–24.

Reul, C., Wick, C., Springmann, U., & Puppe, F. (2017). Transfer learning for OCRopus model training on early printed books. *027.7 Journal for Library Culture*, *5*(1), 38–51.

Rigaud, C., Doucet, A., Coustaty, M., & Moreux, J.-P. (2019). ICDAR 2019 competition on post-OCR text correction. In *2019 International Conference on Document Analysis and Recognition (ICDAR)* (pp. 1588–1593).

Springmann, U., Fink, F., & Schulz, K. U. (2016). Automatic quality evaluation and (semi-) automatic improvement of OCR models for historical printings. *ArXiv*

e-prints. Retrieved from `http://arxiv.org/abs/1606.05157`

Springmann, U., Reul, C., Dipper, S., & Baiter, J. (2018). *Ground Truth for training OCR engines on historical documents in German Fraktur and Early Modern Latin.*

Wemhoener, D., Yalniz, I. Z., & Manmatha, R. (2013). Creating an improved version using noisy OCR from multiple editions. In *2013 12th International Conference on Document Analysis and Recognition* (pp. 160–164).

Wick, C., Reul, C., & Puppe, F. (2018). Calamari-A high-performance tensorflow-based deep learning package for optical character recognition. *arXiv preprint arXiv:1807.02004*.

Xu, S., & Smith, D. (2017). Retrieving and combining repeated passages to improve OCR. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)* (pp. 1–4).

11 Appendix

Here we describe the full set of standard features for the re-ranking step in A-PoCoTo in the initial setting before the start of this project (cf. Section 2.2).

- The *agreeing candidate feature* counts the number of tokens in the multiset $\{w_{mocr}, w_{slocr}\}$ that agree with the correction suggestion $w_{cand,hist}$.
- The *OCR Levenshtein distance feature* yields the Levenshtein-distance between the w_{mocr} and w_{slocr}.
- The *unigram candidate feature* yields the relative frequency of the correction suggestion in the master-OCR document.
- The two Boolean *OCR match features* respectively indicate if w_{mocr} and w_{slocr} match $w_{cand,hist}$.
- The *OCR-pattern confidence features* give the confidence value for w_{mocr} as given by the master OCR-engine.
- The binary *agreeing OCR feature* counts the number of tokens in $\{w_{slocr}\}$ that agree with w_{mocr}.
- The two *unigram OCR features* respectively yield the relative frequency of w_{mocr} and w_{slocr} in the complete master-OCR document.
- The *OCR-token length features* yield the length of w_{mocr} and w_{slocr}.
- The two *trigram features* respectively yield the probability of w_{mocr} and w_{slocr} with respect to an external character-trigram language model obtained from a historical corpus.

- The *profiler weight feature* yields the profiler's weight for the correction suggestion.
- The *profiler weight difference feature* yields the difference of the profiler's weight for the current correction suggestion and the next-ranked correction suggestion.
- The two *OCR maximum character confidence features* yields the maximal confidence value of any character of w_{mocr} and w_{slocr} respectively given by the master- and slave-OCR.
- The two *OCR minimal character confidence features* yield the parallel minimal confidence values.