Pattern Recognition 46 (2013) 1346–1357

Contents lists available at SciVerse ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Unsupervised profiling of OCRed historical documents

Ulrich Reffle, Christoph Ringlstetter *

University of Munich, Center of Information and Language Processing, Germany

ARTICLE INFO

Article history:
Received 19 March 2012
Received in revised form
15 September 2012
Accepted 1 October 2012
Available online 9 November 2012

Keywords:
Error detection and error correction
Processing of historical documents
OCR postprocessing
Statistical learning

ABSTRACT

In search engines and digital libraries, more and more OCRed historical documents become available. Still, access to these texts is often not satisfactory due to two problems: first, the quality of optical character recognition (OCR) on historical texts is often surprisingly low; second, historical spelling variation represents a barrier for search even if texts are properly reconstructed. As one step towards a solution we introduce a method that automatically computes a two-channel profile from an OCRed historical text. The profile includes (1) "global" information on typical recognition errors found in the OCR output, typical patterns for historical spelling variation, vocabulary and word frequencies in the underlying text, and (2) "local" hypotheses on OCR-errors and historical orthography of particular tokens of the OCR output. We argue that availability of this kind of knowledge represents a key step for improving OCR and Information Retrieval (IR) on historical texts: profiles can be used, e.g., to automatically finetune postcorrection systems or adapt OCR engines to the given input document, and to define refined models for approximate search that are aware of the kind of language variation found in a specific document. Our evaluation results show a strong correlation between the true distribution of spelling variation patterns and recognition errors in the OCRed text and estimated ranks and scores automatically computed in profiles. As a specific application we show how to improve the output of a commercial OCR engine using profiles in a postcorrection system.

1. Introduction

As a result of Google Books [1] and many mass digitization projects in libraries around the world, the universe of documents available in search engines and digital libraries is currently changing. Until recently, almost all documents were "born digital", containing modern language. In the wake of the above projects, huge collections of OCRed historical documents have become available, and each month new collections are added. Looking at the contents covered, these collections constitute an important part of the cultural heritage and historical knowledge of the world. However, access to these documents often remains difficult [2–4]. Two core problems are low quality of OCR results and the historical language barrier.

Despite of recent progress [5], even today the quality of OCRed historical texts is often low [6,7]. This is due to several reasons. Historical fonts often change their form from book to book and are difficult to read. The quality of the paper and the images of historical documents is often suboptimal due to noise and geometric distortion [8], and linguistic components of current OCR systems are often not aware of the kind of language variants found in historical texts [9]. Resulting low recognition rates affect the quality of information retrieval systems for historical documents [10] and sometimes lead to complete text regions unacceptable for the human readers [11].

Even if historical texts are recognized perfectly, there is no simple way of how to relate user queries, formulated in modern language, with historical language found in the documents. Due to missing standardization, an immense number of distinct orthographic variants for modern words is typically found in old texts. Matching-based approaches can help to "translate" historical spellings into modern language and vice versa. However, to fine-tune matching, the specific kind of orthographic variation found in a document should be taken into account. As a further complication it also must be noted that OCR errors and historical spelling variation can not be considered in isolation when providing access to OCRed historical texts.

In practice, the spectrum of OCR errors and the specific kind of historical orthographic variation found changes from document to document. Hence, when dealing with the above problems, approaches that are sensitive to the specific properties of the individual input document are of central interest. Looking at the complete processing pipeline from OCR recognition and postcorrection to document indexing and querying, progress can be expected from methods that use a maximum amount of knowledge on each individual input document. As a second requirement, in the context

* Corresponding author. Tel.: +49 8921809715.
E-mail address: kristof@cis.uni-muenchen.de (C. Ringlstetter).

0031-3203/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.patcog.2012.10.002

of "industrial" mass digitization, unsupervised and fully automated approaches are needed. In practice, suitable knowledge that can be used in the above pipeline typically does not come directly as a part of the input. In this situation, since input documents are often long, it is natural to recognize a representative part of the given document with a first run of the OCR engine, and to analyze this intermediate result in a careful and systematic way in order to derive a maximum amount of useful knowledge on the given input document.

In this paper we introduce a method for profiling a given OCRed historical text in a fully automated way. The profile computed for an individual OCRed text comes with global and local information. The *global profile* provides (i) a list of typical recognition errors in the OCR output with an estimate of the number of occurrences of each type of error, (ii) a list of typical patterns for historical spelling variation found in the document with an estimate of the number of occurrences of each type of pattern, (iii) a special kind of language model for the input text, and other useful data. The *local profile* provides a ranked list of individual "interpretations" for each single token of the OCR output. An interpretation of a token represents a hypothesis on the corresponding correct word in the ground truth and on its modern spelling. Intuitively, the notion of an interpretation is based on a two-channel model where a modern text is first rewritten to historical language (historical spelling channel, leading to the OCR input document) and then garbled by OCR (OCR channel, leading to the OCR output document which serves as the input of the profiler). From an algorithmic point of view, our computation of profiles can be considered as a special form of an expectation–maximization procedure where global and local profiles are improved using an iterative mutual reinforcement principle. To guarantee efficiency of the computation, special finite-state technology has been developed [12].

Obviously, profiles of this form offer many interesting possibilities for adaptation of OCR engines, for postcorrection and for information retrieval. For example, the list (i) of typical recognition errors with the estimate on the number of occurrences, as well as the characterization of particular OCR errors in single tokens derived in local profiles could be used to improve the symbol classifiers of an adaptive OCR engine. In postcorrection, the same kind of information can be used to point to suspicious recognition results, and to fine-tune the generation and ranking of plausible correction suggestions for ill-formed tokens. The list (ii) of historical spelling variants found in the document, the text model and other local or global information on the language found in the document could be used to improve the linguistic component of an adaptive OCR engine or to synchronize query expansion of an information retrieval system with the expected historical variants in the document. For all these applications it is important to note that all information is derived in a fully automated way from the initial OCR output. Hence no manual intervention is needed.

All information collected in a profile represent probabilistic estimates, or hypotheses. Our evaluation results on a collection of OCRed historical documents show that estimates collected in global profiles in most cases come close to the true distribution of OCR errors and historical spelling patterns found in the text. In addition, local profiles often offer correct interpretations for the particular tokens of the OCRed text. After this form of base evaluation we give two immediate practical applications of the method in a postcorrection context. We show how the profiles improve *k*-best ranking of correction suggestions for OCR errors on a collection of documents from three centuries, and we report on an experiment where profiling information is used to simplify and speed-up manual postcorrection of OCR results. By offering various batch modes for detecting and correcting possible OCR errors, correction speed is improved in a significant way and the amount of necessary human work is reduced.

The paper is structured as follows. In Section 2 we describe related work. In Section 3 we describe the typical background scenario where we want to apply profiling of OCRed historical texts and introduce some terminology. Section 4 explains how to compute interpretations (in the above sense) for the tokens of the OCR output. This subprocedure is an important ingredient of the profiling algorithm. We also show how to assign a probability to each interpretation. Section 5 describes the profiling method, which can be considered as a special kind of expectation–maximization procedure. Section 6 presents evaluation results where we compare estimates computed in profiles with empirical values obtained from ground truth. In addition we report on a *k*-best ranking experiment for correction candidates and possible improvements of IR access. Section 7 describes the application of the technology in a postcorrection scenario. We finish with a conclusion in Section 8 where we also point to future work and other applications of the technology.

2. Related work

We are not aware of related work considering a two-channel model which looks at historical language variation and OCR errors at the same time. Related papers on processing of historical language have already been cited above. Here we refer to previous work with a focus on modeling recognition errors. Shannon [13] contributed the source-channel paradigm, on which all subsequent probabilistic approaches rely that try to reconstruct the original from the observable text. The advances in the modeling of the OCR channel often could benefit from ideas developed earlier in the field of spelling correction systems and speech recognition. Already in their paper on the string-to-string correction problem, Wagner and Fisher [14] assume an arbitrary cost function which assigns to each edit operation a nonnegative real number as an alternative to uniform edit weights. A number of supervised learning approaches have been proposed to learn probabilities or weights that model the character transformations caused by the noisy channel [15–18]. Weigel et al. [15] describe a dictionary based lexical postprocessing approach. The correction candidates are computed using an iterative supervised learning algorithm which determines the costs for the edit operations. Ristad and Yianilos [16] proposed a fully fledged stochastic model for learning string edit distance from a corpus of examples. The approach is exemplified on the pronunciation of words in conversational speech but can be applied to any string edit problem. Brill and Moore [17] presented a new model for noisy channel spelling correction. It allows unrestricted string to string edits that are learned on a corpus of spelling errors aligned with their correct writing. The probabilities for an edit sequence $\alpha \rightarrow \beta$ are estimated from the number of its occurrences in the training corpus divided by an estimate of all α sequences in a reference corpus. In Kolak et al. [18] a generative model for OCR is introduced with the intention to improve postcorrection. The character sequence transformation is implemented as probabilistic string edit process which is trained by a Viterbi model using a corpus of training examples. To the best of our knowledge the first system directed towards unsupervised learning is the spelling correction system presented by Church and Gale [19]. In the beginning, all edit operations are assumed to be equally probable. From a large text corpus all strings are retrieved that are not in the background lexicon and that lead to a valid word with one edit operation. Iteratively the spelling system runs over the string collection and then uses the corrections made to update the edit probabilities held in character confusion matrices.

Some previous attempts have been made to skip the questionable assumption to have generic training materials for OCR: the characteristics of an individual OCR document depend substantially on the settings of the recognition process, image quality, fonts, etc. [20,21]. Tong and Evans [22] presented a postcorrection system for OCR results based on statistical language modeling which exploits document centric character confusion probabilities. These confusion probabilities are modeled assuming character recognition as an independent process which involves for each character the three basic edit operations. As they point out, the basic probabilities can be estimated according to e.g. the number of substitutions of a character divided by number of occurrences of this character in the groundtruth text, if a training corpus would be available. However, since the character confusion characteristics are dependent on a set of features which changes basically for each text they neglect this possibility. Instead they propose to exploit an iterative learning from correcting technique similar to that in [19] where the system's output of each round is used to learn the confusion probabilities by comparison with the original OCR output. These probabilities are then used in the next iterative step. In [23] we investigated how a fixed strategy for correcting non-lexical tokens appearing in OCR results can be improved by computing an individual error profile for the text. The basic strategy can be sketched as follows. For each word w of the OCR result *not found in the dictionary*, a small list of correction candidates are generated using the dictionary. Each correction candidate v receives a score based on the Levenshtein distance between w and v and the frequency of v. The word w is replaced by the correction suggestion v with the best score if this score exceeds a given threshold. Using the error profile the basic strategy is refined by replacing the Levenshtein distance (which has a uniform cost of 1 for all edit operations) by a variant with symbol dependent edit weights. Weights for symbol dependent edit operations are derived from the static error profile. In Jin et al. [24] the expectation–maximization algorithm is used to adopt an initially equal likelihood distribution of correction candidates. After the first round the distribution is used to estimate a term frequency distribution of the document. The iteration is carried out until the both distributions converge. The model underlies the simplifying assumption that the conditional probability of a word w resulting in a specific OCR token only depends on its rank position in the correction list.

3. Background scenario and terminology

We consider a scenario where we analyze the output document of an OCR engine, assuming that the input document contains historical language. The input document comes with a fixed base language (e.g. English or German) and can contain special vocabulary from other languages (e.g., Latin, French).

In what follows, OCR tokens, defined as OCR output separated by whitespace, punctuation marks or line breaks, are denoted w_{ocr}. When ignoring segmentation errors, word merges and splits, a token w_{ocr} corresponds to a well-defined word w_{gt} in the ground truth (that is, the original, error-free) version of the underlying document. As a matter of fact, we do not know w_{gt} in practical applications. When we "guess" the ground truth version (s.b.) we write w_{candgt}. By an *ocr-trace* we mean a formal description τ_{ocr} which states which kind of OCR errors occurred at which positions of w_{gt} (resp. w_{candgt}), resulting in w_{ocr}. We write

$$w_{gt} \xrightarrow{\tau_{ocr}} w_{ocr}$$

or

$$w_{candgt} \xrightarrow{\tau_{ocr}} w_{ocr}.$$

The former (latter) notation refers to a valid (guessed) relationship. For the explicit notation of traces we use square brackets. For example, $[(u \mapsto ii,2),(l \mapsto t,5)]$ is an OCR-trace which explains how w_{gt}="bubble" is misrecognized as w_{ocr}="biibbte".

The OCR input document contains historical language. Hence a word w_{gt} in the ground truth version of the document will often *correspond* to a word w_{mod} of modern language in the sense that w_{mod} represents the correct modern spelling of w_{gt}. For example, the word "Thurm" (English: tower) of historical German corresponds to the modern word "Turm". In the simplest case, w_{gt} is a modern word, which means that $w_{gt} = w_{mod}$. In many other cases, however, modern and historical spelling are distinct. Special lexica [6] or matching-based approaches [25,26,12] can be used to find the modern spelling of a word found in a historical text. The latter approaches are typically based on a set *Pat* of "patterns" (rewrite rules) such as "t ↦ th" that locally explain the difference between modern and historical spelling.[1]

When we consider a word w_{gt} occurring in the ground truth version and ignore all contextual information, there are sometimes several modern words $w_{candmod}$ that "might" correspond to w_{gt}. We assume that the correspondences between a historical spelling and a modern word can be described by means of a formal description τ_{hist} which explains the derivation of w_{gt} from w_{mod}. A description of this form is called a *hist-trace*. Typically, a hist-trace lists rewrite patterns and the positions where they have been applied. We also use one special form of hist-trace which just states that the association between a historical word and a modern pendant is "irregular", which means that there is no pattern-based derivation. Using similar notational conventions as above we write

$$w_{mod} \xrightarrow{\tau_{hist}} w_{gt}$$

or

$$w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt}.$$

Combining the two processes we arrive at the following notion.

Definition 3.1. An *interpretation* of a token w_{ocr} of the OCR output is a quintuple written in the form

$$w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt} \xrightarrow{\tau_{ocr}} w_{ocr}$$

where

- w_{candgt} represents a candidate for the ground truth version of w_{ocr}.
- $w_{candmod}$ is a candidate modern word that might correspond to w_{candgt}.
- τ_{hist} is a hist-trace,
- τ_{ocr} is an ocr-trace.

A *ground truth interpretation* has the form

$$w_{candmod} \xrightarrow{\tau_{hist}} w_{gt}$$

and shows how a word of a ground truth text is derived from a modern word using a hist-trace.

Example 3.2. As an example, assume that the string "tneil" is an output token of the OCR which has recognized a historical

[1] In most European languages, for example Dutch, French, English, Bulgarian, Czech and Slovene, to name just a few, a significant part of the historic vocabulary (non-modern words), for German more than 50% of the types and more than 75% of the tokens, can be traced back to simple pattern transformations. The language channel of the profiling technology models a document centric selection of these patterns.

German text. A possible interpretation is

$$teil \xrightarrow{[t \mapsto th,1]} theil \xrightarrow{[h \mapsto n,2]} tneil,$$

where the modern German word "teil" corresponds to the historical spelling variant "theil", which is misrecognized as "tneil".[2] Another possible interpretation, leading to the modern word "keil" (English: quoin, or cotter) is

$$keil \xrightarrow{[\,]} keil \xrightarrow{[(k \mapsto tn,1)]} tneil$$

where $[(k \mapsto tn, 1)]$ describes a split operation produced by the OCR engine.

Depending on the maximal number of rewrite operations that are tolerated in interpretations (s.b.), an OCR output token can have many interpretations, some of which might seem natural, others odd. The OCR output can also contain *exceptional* tokens w_{ocr} that do not have any valid interpretation. This holds if either w_{ocr} does not correspond to any single token of the ground truth version (segmentation errors, word splits and merges) or the ground truth version w_{gt} does not correspond to any modern word in the above sense (being either a modern string or derivable as one by lexical resources or the pattern based matcher). A related question is if our automated methods, which are based on limited resources, are able to derive an interpretation for a given token w_{ocr} (regardless of being the true one, w_{gt}). In the positive case we say that token w_{ocr} is *interpretable*.

In an obvious sense, the notion of an interpretation gives rise to a *two-channel model*. The first channel ("*hist-channel*") describes the mutation of $w_{candmod}$ to w_{candgt} where historical patterns are applied in parallel. The second channel ("*OCR-channel*") describes the mutation of w_{candgt} to w_{ocr} where error patterns (edit operations) are, again, applied in parallel.

4. Generating and weighting interpretations

Before we introduce our method for profiling OCRed historical texts, we show how different lexica in combination with approximate matching procedures are used to generate a meaningful set of possible interpretations for all interpretable tokens of the OCR output (Section 4.1). Once the set of interpretations for a token is computed, probabilities are estimated for each interpretation, as we explain in Section 4.2. This computation of interpretations and probabilities can be considered the core subprocedure of the profiling algorithm.

4.1. Computing interpretations for OCR tokens

Resources: For the computation of interpretations for all tokens w_{ocr} of the OCR output distinct types of language resources can be used. A minimal setup consists of only two resources:

1. a lexicon L_{mod} of full forms in their modern spelling for the base language of the input document,
2. a set of patterns, *Pat*, capturing the historical spelling variation for the base language.

We then use a special matching procedure to compute interpretations for the tokens w_{ocr}. In [12] we introduced an efficient procedure for *variant-aware approximate search* in dictionaries: given the above resources, the user may specify bounds for the maximal number of operations to be tolerated in both the hist- and the ocr-trace. The complete sets of possible OCR errors and historical patterns can be declared in a configuration file. The algorithm then generates all interpretations for w_{ocr} that satisfy these bounds. Depending on how restrictive the bounds are chosen, the generated result set will also include interpretations that are not quite plausible. Concerning the *hist-channel*, more precision can be expected if specific resources for historical language are available. Complementing the matching approach mentioned above, our configuration for historical German included two different lexica with non-modern vocabulary:

3. A lexicon $L_{histcorp,manual}$ of *non-modern* historical word forms from a proofread historical corpus where lexicographers had assigned corresponding modern word forms and hist-traces to the entries.
4. A lexicon $L_{histcorp,matching}$ of additional modern and historical word forms from the same historical corpus. Possible corresponding word forms and (empty or non-empty) hist-traces were pre-computed using the matching procedure mentioned above. In the hist-traces, at most two pattern applications were tolerated.[3]

Note that interpretations $w_{candmod} \overset{\tau_{hist}}{\rightarrow} w_{candgt}$ in resource 3 are manually approved. In resource 4, parts $w_{candmod}$ and τ_{hist} are not manually verified, but the ground truth token w_{candgt} has empirical evidence from corpus data. $L_{histcorp,manual}$ as well as $L_{histcorp,matching}$ may contain ambiguous entries, i.e. suggest more than one ground truth interpretation for one w_{ocr}. Also note that resources 1, 3 and 4 are disjunct.

Finally, additional resources were used to cover special vocabulary that is usually not found in conventional full form lexica.

5. Special lexica for Latin, geographic names and person names.

Exact lookup in lexica: In order to generate interpretations for an OCR token w_{ocr}, we first try and see if w_{ocr} can be found in the modern lexicon L_{mod}. If this is the case, we produce exactly one trivial interpretation where $w_{candmod} = w_{candgt} = w_{ocr}$ and both the hist-trace and the ocr-trace remain empty

$$w_{ocr} \overset{[]}{\rightarrow} w_{ocr} \overset{[]}{\rightarrow} w_{ocr}$$

Next we look for exact matches of w_{ocr} in the historical lexica ($L_{histcorp,manual}$ and $L_{histcorp,matching}$). Recall that these lexica assign to historical word forms w_{candgt} complete ground truth interpretations of the form $w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt}$. All entries where w_{candgt} matches w_{ocr} are added to the set of interpretations, using an empty OCR-trace

$$w_{candmod} \xrightarrow{\tau_{hist}} w_{ocr} \xrightarrow{[]} w_{ocr}$$

This exact lookup procedure covers all words w_{ocr} that are present in one of our lexica. We consider all those words to be correctly recognized by the OCR, ignoring the problem of real-word errors ("false friends") at this point. Accordingly, all generated interpretations have an empty OCR-trace.

Query for approximate matches: Only if the exact lookup does not yield any results, we apply approximate search strategies to find interpretations for w_{ocr}. We use the technology described in [27] to find all entries of L_{mod}, $L_{corpus,manual}$ and $L_{corpus,matching}$ where the standard Levenshtein distance[4] between w_{ocr} and w_{candgt} does

[2] Because of inconsistencies for Early New High German, capitalization of nouns is a soft feature in the system.

[3] For more than two pattern applications the wide majority of the predicted modern–historical word pairs turned out to be inconsistent.

[4] The standard Levenshtein distance between two strings v and w is the minimal number of letter insertions, deletions, or substitutions needed to rewrite v into w.

not exceed 2. The respective ocr-traces are computed using standard dynamic programming techniques. For the special lexica, e.g. Latin and geographic names, we specify a threshold of standard Levenshtein distance 1.

In order to find the correct interpretations also for historical variants w_{gt} that are not covered by our corpus-based lexica, *variant-aware approximate search* is applied using L_{mod} together with the set of patterns *Pat*. Here, the search is restricted to 2 variant pattern applications and standard Levenshtein distance 1.

Remark 4.1. When treating documents from other base languages, we cannot always expect to have the same kind of language resources as those described above. As pointed out at the beginning of this subsection, as a minimum requirement for the profiling technique we need a lexicon of modern words for the base language and a set of patterns that captures historical spelling variation. In this case, the simplified procedure for the generation of interpretations consists of exact matching in L_{mod} and then variant-aware approximate matching using L_{mod} and *Pat*.

4.2. Estimating probabilities for interpretations

As a last preparation for the description of the profiling algorithm we now show how to estimate probabilities for each interpretation assigned to a given OCR output token. Consider an interpretable OCR output token w_{ocr}. Using Bayes rule we obtain

$$P(int|w_{ocr}) = \frac{P(w_{ocr}|int)\cdot P(int)}{P(w_{ocr})}$$

for each interpretation *int* of w_{ocr}. Strictly speaking we are able to estimate $P(int|w_{ocr})$ only up to the unknown constant $P(w_{ocr})$. However, when we assume that our list of candidate interpretations for w_{OCR} contains all possible interpretations (a simplification), then we can

1. in a first step compute the "weight" $w(int) := P(w_{ocr}|int)\cdot P(int)$ for each candidate interpretation,
2. then distribute the "probability mass 1" among all possible interpretations in a way proportional to weights to obtain probabilities $P(int|w_{ocr})$.

It remains to estimate weights $w(int) = P(w_{ocr}|int)\cdot P(int)$. Let

$$int = w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt} \xrightarrow{\tau_{ocr}} w_{ocr}.$$

Since the interpretation explains "how w_{ocr} was born" we have $P(w_{ocr}|int) = 1$, which shows that weights are given by the unconditional probabilities

$$w(int) = P(w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt} \xrightarrow{\tau_{ocr}} w_{ocr}).$$

Assuming that in both channels distinct patterns resp. OCR errors are applied in an independent manner we obtain

$$\begin{aligned} w(int) &= P(w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt})\cdot P(\tau_{ocr}) \\ &= P(w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt})\cdot \prod_{err\in\tau_{ocr}} P(err) \\ &= P(w_{candmod})\cdot P(\tau_{hist})\cdot \prod_{err\in\tau_{ocr}} P(err) \\ &= P(w_{candmod})\cdot \prod_{pat\in\tau_{hist}} P(pat) \prod_{err\in\tau_{ocr}} P(err) \end{aligned}$$

Here *err* (*pat*) denotes an OCR error pattern (pattern for historical language variation) occurring in the trace τ_{ocr} (τ_{hist}).

$P(w_{candmod})$ is the probability that a word randomly selected from a historical corpus in its modernized spelling has the form $w_{candmod}$. $P(pat)$ is the probability that the left-hand side of *pat* – as a substring of a modern word – is rewritten to the right-hand side in a corresponding historical spelling variant. Similarly $P(err)$ is the probability that the left-hand side of *err* – as a substring of a word of the OCR input document – is rewritten to the right-hand side by the OCR engine. Obviously, the factors $P(w_{candmod})$, $P(pat)$, and $P(err)$ are exactly the kind of information that is provided by the following type of probabilistic model.

Definition 4.2. A *text-channel-model* for an OCRed input text *T* is a triple $Mod = (V,O,H)$ where

1. *V* is a language model describing the probability of a modern word to "occur in some spelling" in the ground truth version T_{gt} of *T*. By a modern word we mean an entry of the modern lexicon L_{mod} or an entry of a special lexicon. We say that *w* "occurs in some spelling" in the text T_{gt} if *w* is the modern pendant of a spelling variant found in T_{gt}.
2. *O* is a probability distribution that specifies a set of edit operations that can occur as OCR errors and defines for each such edit operation a probability. *O* is meant to describe the OCR-channel leading from T_{gt} to *T*.
3. *H* is a probability distribution that assigns to each pattern *pat* in the set of patterns *Pat* a probability $P(pat)$. *H* is meant to describe the distribution of patterns found in the language of the historical text T_{gt}.

Note that a text-channel-model is a simple form of a "global profile" as described in the Introduction. In general, global profiles can contain additional data.

Summing up we have seen that estimates for the probability of the distinct interpretations assigned to a given token of the OCR output text can be derived directly from a suitable text-channel-model for the input text.

5. Profiling OCRed historical texts

In this section we describe the profiling of OCRed historical texts. We start with an overview of the algorithm, which represents a special type of expectation–maximization procedure. In the following subsections we provide missing details. We also comment on the language resources that are needed for each step.

5.1. Overview

The global structure of the algorithm for profiling a given OCRed historical text can be described as follows.

Initialization: In an offline step, independent of the concrete input texts, we compute an initial "rough" text-channel-model $Mod_0 = (V_0,O_0,H_0)$. V_0 and H_0 – the components of the model which refer to the language – are estimated using a corpus of non-annotated texts.[5] O_0, the initial distribution of OCR error patterns, assigns uniform weights to all possible OCR error patterns. We treated letter insertions, deletions, single-letter substitutions, 2 : 1 letter merges and 1 : 2 letter splits as equally weighted possible OCR errors.

Round 1: Using the initial text-channel-model Mod_0 we compute for each token of the OCR output a set of possible interpretations with probabilities (*first local profile*). Collecting hypotheses from each single token and accumulating this information we obtain an improved text-channel-model Mod_1 which represents the *first global*

[5] Non-annotated texts come as raw texts without information e.g. on modern–historical variants.

profile of the text. With the computation of this improved text-channel-model, which now is "text centric", Round 1 is finished.

Iteration: We then iterate two steps. In Step 1, the text-channel-model Mod_i for the specific text obtained after Round i is used to derive for each token of the OCR output an improved set of possible interpretations with probabilities (*local profile* $i+1$). In Step 2, collecting hypotheses from each single token and accumulating this information we obtain an improved text-channel-model Mod_{i+1} which represents the *global profile* $i+1$ of the text.

Termination, output: Experience shows that usually the text-channel-model converges after at most six iterations. The output consists of the final local and global profiles.

Remark 5.1. Obviously, Round 1 is just a special case of the Iteration step. In our description we treated this as an extra step just to point to the fact that in Round 1 we proceed from a text-channel-model which is independent from the input text to an improved model for the specific text.

Remark 5.2. In general we will not find a meaningful interpretation for each token of the OCR output text. As mentioned before, we say that a token w_{ocr} of the OCR output is *interpretable* if our automated method has at least one interpretation for w_{ocr}. Refining the above picture, local profiling is restricted to interpretable tokens of the OCRed input text.

5.2. Initialization

For the computation of a basic text-channel-model for arbitrary OCRed input texts we use a large non-annotated historical corpus[6]. We use the matching procedure described in [12] (cf. Section 4.1) to automatically assign a ground truth interpretation $w_{candmod} \xrightarrow{\tau_{hist}} w_{gt}$ to each word w_{gt} of the corpus. Where the result of the matching procedure is ambiguous, we use the interpretation which is minimal with respect to the number of applied variant patterns in τ_{hist}.

Estimating the probability of historical patterns: The traces found in the interpretations of the historical corpus can be used to estimate how often a pattern *pat* has been applied in the corpus. Let $n_{hist}(pat,1)$ denote this number. We can also compute the number $n_{hist}(pat,0)$ of occurrences of the left-hand side of the pattern that were *not* replaced by the right-hand side of *pat*. We can now estimate

$$P(pat) = \frac{n_{hist}(pat,1)}{n_{hist}(pat,1)+n(pat,0)}$$

Estimating the probability of modern words: As a first approximation we may estimate the probability $P(w_{candmod})$ of a modern word by counting how often $w_{candmod}$ occurs in the ground truth interpretations of the background corpus and dividing this number by the total number of tokens of the corpus. However, when profiling an OCRed document, correct interpretations will often lead to modern words that do not occur in the interpretations of the training corpus. To provide a very simple smoothing, we assign frequency 1 to each unknown $P(w_{candmod})$ and derive a probability as explained above.

Estimating probabilities for OCR errors: The estimation of the probability of OCR errors is notoriously difficult. As mentioned before, in the current version we initially allow all possible insertions, deletions, single-letter substitutions, 2 : 1 letter merges and 1 : 2 letter splits. All error patterns are assigned a uniform probability.

5.3. Iteration, computation of local profiles

As we have seen in Section 4.1, the profiler uses a subprocedure where for each interpretable token w_{ocr} of the OCRed text a list of candidate interpretations is generated. In Round $i+1$ we use the text-channel-model Mod_i obtained from Round i to re-estimate the probability for each interpretation of a token w_{ocr} as described in Section 4.2. The updated model Mod_i also feeds into the computation of the OCR-traces in interpretations—for some pairs (w_{candgt},w_{ocr}), new weights for OCR errors might lead to a different sequence with minimal costs.

5.4. Iteration, computation of global profiles

We now describe the second step of the expectation–maximization Procedure, where the information from the local profiles of the previous iteration is accumulated to obtain an improved global profile, i.e. a new text-channel-model with refined probabilities. To this end, we use counters similar to the initialization described in Section 5.2: $n_{hist}(\alpha \mapsto \beta,1)$ denotes how often the spelling variant pattern $\alpha \mapsto \beta$ has been applied–$n_{hist}(\alpha \mapsto \beta,0)$ denotes the number of occurrences of α where the pattern has *not* been applied. An equivalent counter n_{ocr} is used to count occurrences of OCR patterns. Finally, $n_{wmod}(w)$ is used to count how often w appears as w_{mod} of a word in the document.

In our case, however, "counting occurrences" is not as entirely trivial as it may appear: the local profile for a given word does not provide a definite interpretation, instead it contains a set of candidate interpretations with respective probabilities. So, all interpretations contribute to the counters with their probability share. For every candidate interpretation of the form

$$w_{candmod} \xrightarrow{\tau_{hist}} w_{candgt} \xrightarrow{\tau_{ocr}} w_{ocr}$$

appearing in the local profile with probability p, we increment the counters as follows:

- For all variant patterns *pat* appearing in τ_{hist} we add p to $n_{hist}(pat,1)$.
- For all OCR error patterns *err* appearing in τ_{ocr} we add p to $n_{ocr}(err,1)$.
- For all substrings α in $w_{candmod}$ where the variant pattern $pat = \alpha \mapsto \beta$ was not applied we add p to $n_{hist}(pat,0)$.
- For all substrings α in w_{candgt} where the OCR error pattern $pat = \alpha \mapsto \beta$ was not applied we add p to $n_{ocr}(pat,0)$.
- Finally, we add p to $n_{wmod}(w_{candmod})$.

Having accumulated this information from the local profile, we can now compute new estimates for the text-channel-model.

Re-estimating probabilities for variant patterns: The new estimate for the probability of the variant pattern *pat* is the quotient

$$P(pat) = \frac{n_{hist}(pat,1)}{n_{hist}(pat,1)+n_{hist}(pat,0)}$$

Re-estimating probabilities for OCR errors. The new estimate for the probability of the OCR error pattern *err* is the quotient:

$$P(err) = \frac{n_{ocr}(err,1)}{n_{ocr}(err,1)+n_{ocr}(err,0)}$$

In order to keep the set of OCR error patterns from getting unreasonably large, we drop error patterns if they do not appear in the top interpretation of at least two words of the document. For all patterns that are not assigned a specific probability a smoothing value can be specified.

Re-estimating the probabilities for modern words: We estimate the probability for a word w to be the modern spelling of a word

[6] Because token context is not taken into account, also a type-frequency list serves the purpose. If no historical corpus is available, the historical patterns are treated as equally distributed for initialization.

in the document with

$$P(w) = \frac{n_{wmod}(w)}{N}$$

where N is the total number of words in the document. For w with $n_{wmod}(w) = 0$ we use frequency 1 as a simple smoothing.

5.5. Termination, output

Our experiments showed that usually the text-channel-model converges quickly. After at most six iterations no significant changes could be observed. The output consists of the final local and global profiles. The final local profile, as any intermediate local profile, contains a set of interpretations for each token of the OCR output with probabilities. The final global profile contains, most importantly, the text-channel-model reached after termination. Additionally, it contains statistics on the coverage of the distinct lexica and an estimate of the OCR error rate for the given input text. Other additional text characteristics can be added or deduced from the available data.

Following this procedure, we obtain a text-channel-model that is tailored specifically to the input document.

6. Evaluation

In this section we evaluate the profiling technology on a collection of OCRed historical documents. We first report on the global correlation between the (i) estimated and (ii) true distribution of OCR error types and historical variant patterns, respectively. Then we evaluate the predictive quality of the OCR error model with respect to the ranking of correction candidates for garbled OCR tokens. Two standards of data are used for the evaluation.

Gold standard: For all test documents, the ground truth versions of the OCRed texts were at our disposal. Ignoring effects caused by word segmentation problems, we assigned to each token w_{ocr} of the OCR output the corresponding token w_{gt} of the ground truth. However, this information is not sufficient to compute the optimal "true" profile for the OCR text as a reference point for the evaluation. For the evaluation of local profiles we need to know for each token w_{ocr} the full correct interpretation, which includes the modern pendant w_{mod}, the OCR-trace τ_{ocr}, and the hist-trace τ_{hist}. In what follows, a document where we have for each token w_{ocr} the true interpretation fully available is called a *gold standard*. For manual preparation of gold standards we developed a graphical user interface. Given the OCR output document and the ground truth alignment, for each aligned pair (w_{ocr},w_{gt}) suggestions for the unknown components w_{mod}, τ_{ocr}, and τ_{hist} are automatically generated and selected/corrected by the annotator. Obviously, once we have the perfect local profile for each token we can derive the true global profile. Using the graphical user interface we created gold standards for two test documents from the 18th century. The first document, denoted $G1$, contains approximately 14,400 tokens, the second document $G2$ contains 4,300 tokens.

Silver standard: The preparation of documents in gold standard quality is very time consuming. To have a broader empirical basis for the evaluation of profiles we used an additional collection of test documents. Again the automatic alignment of OCR tokens w_{ocr} and tokens w_{gt} of the ground truth versions of the texts was used as a starting point. Given the pairs (w_{ocr},w_{gt}) and using standard matrix based dynamic programming methods we automatically computed OCR-traces. Manual inspection showed that the accuracy of this method is very high, almost all OCR-traces were correct. A document where each token w_{ocr} comes with the correct word w_{gt} and an OCR-trace computed automatically is called a *silver standard*. Silver standard documents were only used to evaluate the OCR-part of the profiles. We created 15 documents in silver standard, five documents each from the 16th, 17th, and 18th century with together 90,000 tokens.

Resources used for the experiments: For the evaluation experiments we used a lexicon L_{mod} for the modern base language with over 2 million entries and a set of 140 patterns, *Pat*, capturing the historical spelling variation for the base language (German). In addition we had the following resources at our disposal: a lexicon $L_{histcorp,manual}$ with 30,000 (non-modern) historical word forms with manually attached hist-traces. A lexicon $L_{histcorp,matching}$ of 200,000 additional modern and historical word forms from a historical corpus. Corresponding modern word forms and (empty or non-empty) hist-traces were pre-computed using the special matching procedure mentioned above. In the hist-traces, at most two pattern applications were tolerated. Furthermore we used special lexica for Latin (273,723), prominent geographic names (30,262) and prominent person names (8,177).

6.1. Evaluation of global profiles

First we evaluated the global profile of a document, which includes the lists of historical variant patterns and the list of error patterns. Both come as ranked lists together with estimated probabilities from the expectation–maximization procedure. To obtain a general picture of the correlation between predicted models and truth we used the Pearson correlation coefficient. In what follows, the variable X models the estimated distribution of all OCR error patterns according to the global error profile and Y models the true distribution of OCR error patterns in the profiled document computed from ground truth

$$r = \frac{Cov(X,Y)}{\sqrt{Var(X)}\sqrt{Var(Y)}}.$$

The values of the correlation coefficient r for the 15 books of the silver standard are given in Table 1. For most documents the estimates of the global error profiles and the real error distributions show a strong correlation of 0.75 and above. For two documents, S16.2 and S18.5, the correlation coefficients are only 0.41 and 0.39, respectively. Both documents are problematic for our technology: in document S16.2, many OCR errors of the kind ä-a and ö-o were misinterpreted as historical patterns in the profiles (note that in these cases the interpretation nevertheless gives the correct modern word); in S18.5, which represents a play, the two main character names are wrongly profiled as OCR errors, producing false positives in huge numbers. In an interactive scenario, this effect could easily be avoided by adding the two names to the background lexicon of person names.

In Fig. 1 we show scatter plots for two of the profiled books (S16.5, S17.1) included in the silver standard. The most important observation is that the most frequent error types (large y-coordinate) are found by the profiler (large x-coordinate). Applications will benefit mainly from the knowledge on the most frequent error types. Some OCR errors in fact have hundreds of instances per book. Revealing frequent error types can be exploited, e.g., by adding a feed-back loop to adaptive OCR engines, by improving the ranking of correction candidates, or by batch processing of errors in interactive postcorrection systems.

Table 1
Pearson correlation coefficient r of predicted OCR error model and groundtruth (silver standard) for 15 books from three centuries.

Subcorpus	S.1	S.2	S.3	S.4	S.5
r16th century	0.76	0.41	0.62	0.79	0.73
r17th century	0.88	0.79	0.78	0.94	0.85
r18th century	0.64	0.77	0.68	0.87	0.39

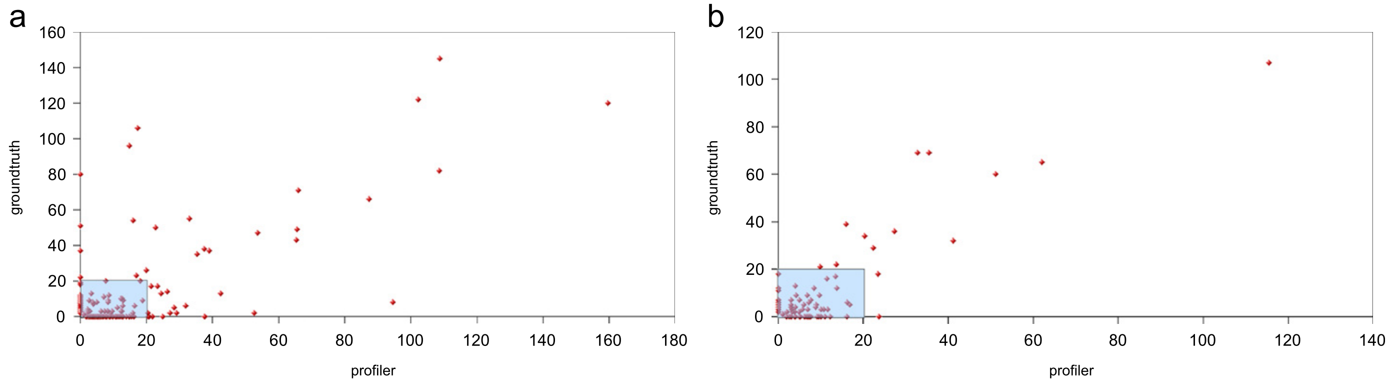
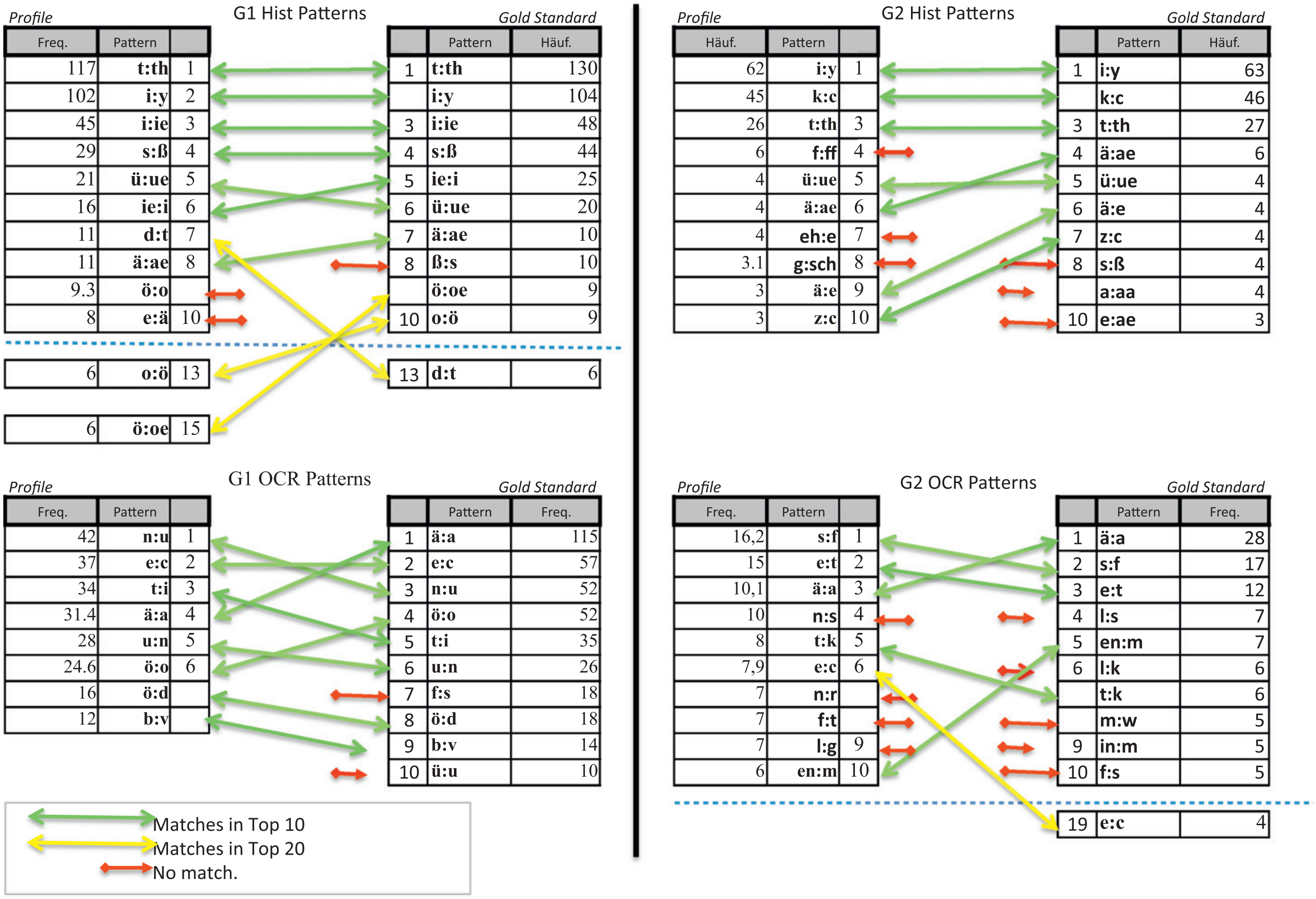


Fig. 1. Scatterplots showing the correlation between estimated OCR error profiles and the actual errors (absolute frequencies). Each point + corresponds to a particular type of OCR error. The *y*-coordinate (groundtruth) gives the true number of occurrences, the *x*-coordinate (profiler) the number estimated via profiling. Since from an application view, the frequent patterns are the main target of the method, the grayed out area below 20 is less important. (a) Scatterplot of document 16.5. (b) Scatterplot of document 17.1.



G1 Hist Patterns

Profile

Freq.	Pattern	
117	t:th	1
102	i:y	2
45	i:ie	3
29	s:ß	4
21	ü:ue	5
16	ie:i	6
11	d:t	7
11	ä:ae	8
9.3	ö:o	
8	e:ä	10
6	o:ö	13
6	ö:oe	15

Gold Standard

	Pattern	Häuf.
1	t:th	130
	i:y	104
3	i:ie	48
4	s:ß	44
5	ie:i	25
6	ü:ue	20
7	ä:ae	10
8	ß:s	10
	ö:oe	9
10	o:ö	9
13	d:t	6

G2 Hist Patterns

Profile

Häuf.	Pattern	
62	i:y	1
45	k:c	
26	t:th	3
6	f:ff	4
4	ü:ue	5
4	ä:ae	6
4	eh:e	7
3.1	g:sch	8
3	ä:e	9
3	z:c	10

Gold Standard

	Pattern	Häuf.
1	i:y	63
	k:c	46
3	t:th	27
4	ä:ae	6
5	ü:ue	4
6	ä:e	4
7	z:c	4
8	s:ß	4
	a:aa	4
10	e:ae	3

G1 OCR Patterns

Profile

Freq.	Pattern	
42	n:u	1
37	e:c	2
34	t:i	3
31.4	ä:a	4
28	u:n	5
24.6	ö:o	6
16	ö:d	
12	b:v	

Gold Standard

	Pattern	Freq.
1	ä:a	115
2	e:c	57
3	n:u	52
4	ö:o	52
5	t:i	35
6	u:n	26
7	f:s	18
8	ö:d	18
9	b:v	14
10	ü:u	10

G2 OCR Patterns

Profile

Freq.	Pattern	
16,2	s:f	1
15	e:t	2
10,1	ä:a	3
10	n:s	4
8	t:k	5
7,9	e:c	6
7	n:r	
7	f:t	
7	l:g	9
6	en:m	10

Gold Standard

	Pattern	Freq.
1	ä:a	28
2	s:f	17
3	e:t	12
4	l:s	7
5	en:m	7
6	l:k	6
	t:k	6
	m:w	5
9	in:m	5
10	f:s	5
19	e:c	4

Fig. 2. Visualization of the global profiles of the two books G1 (left-hand side) and G2 (right-hand side) of the gold standard: green arrows mark the top 10 historical variants (upper parts) and OCR error patterns (lower parts) shared by both profile and groundtruth. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

To give a more detailed and concrete picture on the correlation between estimated and true number of OCR errors (historical patterns) of a particular type, we compare the most important error patterns (profiler versus ground truth). For two fully annotated books of the *gold standard*, called *G*1 and *G*2, in Fig. 2 we show the profiled and the true patterns ordered by frequency.[7] Green arrows mark patterns which belong to the “top 10” for both lists. Yellow arrows mark weaker patterns which in both lists

[7] In the profile, the “frequency” of an error pattern is obtained by accumulating probabilities over all tokens.

reach at least rank 20. Red arrows mark patterns that do not have a partner in the top 20 patterns of the parallel list. Both for historical and error patterns, the list of top 10 types found via profiling is very similar to the true list of top 10 patterns derived from the gold standard. For G1, 7 of the top 10 historical variant patterns suggested by the profiler can be found in the top 10 of the gold standard. Another top 10 patterns have rank 13 in the parallel list. For the OCR error patterns, the first eight patterns found by the profiler belong to the top 10 of the gold standard. Estimated probabilities show a strong correlation with true numbers. An outlier here again is the mutated vowel ä ↦ a. For G2, 7 historical patterns of the profile are found in the true top 10. However, regarding the error patterns, only five patterns belong to the top 10 of the parallel list. One problem with G2 was that the document was rather short and most of the patterns have only 10 hits in groundtruth or less, which makes it difficult for the expectation–maximization algorithm to stabilize.

Table 2 gives an overview for the top 10 OCR error patterns for the *Silver standard*. We measure the overlap between the two lists of 10 most frequent OCR errors as (i) predicted by profiler and (ii) derived from groundtruth. For most documents, the overlap between groundtruth and the estimates is 60% and higher. Again, weaker results obtained for a few documents can be traced back to the unsolved disambiguation problem concerning the mutated vowels. The strong performance of the profiling technology with respect to the top *k* patterns can be exploited for batch processing of prominent errors in a postcorrection system (cf. Section 7).

6.2. Evaluation of local profiles

Ranking of correction suggestions: In order to evaluate the potential of profiling for local decision making we used the local profiles in an experiment where we ranked correction candidates. As in [17] we ran our model on the OCR errors in our evaluation set and measured the *k*-best correction suggestions obtained via profiling against a base line.

We selected all "correctable" OCR errors in the documents of the silver standard where our lexical resources described above were capable to produce the correct word in the list of candidates. For searching correction candidates we used the setting presented above, a historical lexicon and as a fallback solution the approximative lexicon.

The lists of the correction candidates (on average 62.06 per token) were sorted by the document centric error model combined with the available language model based on corpus-frequency as described in the description of profiling above. The results were compared to a baseline model where we used equally distributed error probabilities and the same language model. Tables 3 and 4 show results for both models. Here, *k*-best accuracy means the percentage how often the correct word of the ground truth text is among the top *k* candidates suggested by the method.

The results are clearly superior to the baseline model with a gain of 19.2% for the 16th, 20.2% for the 17th and 14.8% for the 18th century to rank the correct word as the top candidate; respectively gains of 17.1%, 17.1% and 8% to rank the correct word into the first three candidates. Despite a good overall correlation, for document S18.3, which is a comparatively short document with no clear high profile errors and absolute numbers of only 10 −5 occurrences for the recognized top patterns, the technology creates an outlier with 1.4% loss compared to the baseline.

Power of predicting correct modern versions: In an Information Retrieval application, users in most scenarios want to use modern words w in queries. The ideal result set for query w should include all tokens of the OCRed output w_{ocr} where the *corrected and modernized version* w_{mod} of w_{ocr} equals w. When indexing the OCRed text, users only have access (ignoring wrong hits) to correctly recognized words that occur in modern spelling in the text. In the two documents of our gold standard, the percentage of these "accessible" tokens is 85.76% and 72.64%, respectively. When using the profiler, indexing the modern word computed in the best-ranked interpretation of each token, these numbers are raised to 91.25% and 80.77%, respectively.

Table 2
Results for the quality of OCR error profiles of the silver standard, five documents each of the 16th (S16.1,. .,S16.5), 17th (S17.1,. .,S17.5) and 18th century (S18.1,. .,S18.5). We measure the overlap between predicted and true top 10 lists of OCR errors.

16th century	S16.1	S16.2	S16.3	S16.4	S16.5	∅
Top 10 overlap—OCR	60%	40%	50%	40%	60%	50%
17th century	S17.1	S17.2	S17.3	S17.4	S17.5	∅
Top 10 overlap—OCR	80%	60%	60%	70%	70%	68%
18th century	S18.1	S18.2	S18.3	S18.4	S18.5	∅
Top 10 overlap—OCR	40%	60%	50%	70%	50%	54%

Table 3
1-Best ranking of correction candidates using the local error profile against a baseline model using equally distributed errors. Both using a frequency based language model.

*Data Model*1-best						
16th century	S16.1	S16.2	S16.3	S16.4	S16.5	∅
Profiler	63.4	45.1	61.5	52.1	57.7	56.0
Baseline	47.4	29.8	38.6	31.8	36.7	36.8
17th century	S17.1	S17.2	S17.3	S17.4	S17.5	∅
Profiler	74.3	64.3	67.7	76.6	65.9	69.8
Baseline	52.2	51.3	42.7	54.2	47.8	49.6
18th century	S18.1	S18.2	S18.3	S18.4	S18.5	∅
Profiler	72.9	71.2	65.5	67.8	77.3	70.9
Baseline	66.9	55.2	58.4	43.7	56.1	56.1

Table 4
3-Best ranking of correction candidates using the local error profile against a baseline model using Standard Levenshtein Distance. Both using a frequency based language model. S18.3 is a comparatively short document with absolute numbers of high profile errors below 10.

*Data Model*3-best						
16thcentury	S16.1	S16.2	S16.3	S16.4	S16.5	∅
Profiler	81.4	76.4	84.2	76.0	78.2	79.2
Baseline	70.0	56.3	63.8	56.0	64.4	62.1
17th century	S17.1	S17.2	S17.3	S17.4	S17.5	∅
Profiler	94.7	90.1	90.3	92.5	87.6	91.2
Baseline	77.7	80.8	69.3	76.5	66.2	74.1
18th century	S18.1	S18.2	S18.3	S18.4	S18.5	∅
Profiler	89.8	87.7	84.1	89.9	91.9	88.7
Baseline	86.4	81.0	85.8*	68.5	81.9	80.7

Remark 6.1. For German, the profiling method was extensively tested at the Bavarian State Library, building in the profiling technology in a postcorrection tool to speed-up correction (see next section). Corresponding pilot projects for other languages are currently run at the *Bibliotheque Nationale de France*, the *Koninklijke Bibliotheek* of the Netherlands and the *Biblioteca Virtual Miguel de Cervantes* in Alicante. The available resources (size of lexica, number of patterns for historical variants) vary considerably: for example, for French we use a set of 807 variant patterns (we use only 140 for German). From the present view, the profiling method produces good results for all languages mentioned above.

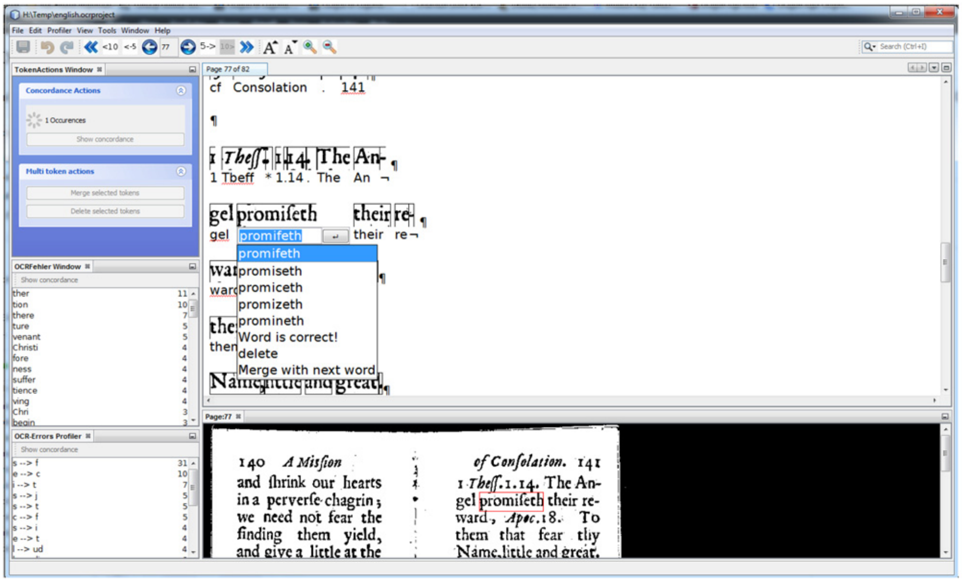


Fig. 3. A screenshot of the postcorrection system with the text view module (1), the complete image view (2), a selected token with correction suggestions (3), the batch view selection for most prominent OCR error patterns recognized by the profiler (4), the batch view selection for most frequent erroneous words predicted by the profiler (5) and a window to correct word segmentation errors (6).

7. Improving postcorrection as an application

The above evaluation experiments show that profiling information could be used at many places in the complete pipeline of full text digitization, from OCR recognition and postcorrection to indexing and querying. One obvious application would be a feedback loop into an adaptive OCR system. Since the effects of adding such a new feature to the core recognition process are complicated, the implementation of such a feedback loop has to be realized at industry level. In our own work we had to treat the given OCR engine as a black box. Here the obvious choice was to integrate the profiling technology into OCR postprocessing. To this end a tool for postcorrection of OCR results has been implemented where dictionaries, text profiles, error profiles and language models are fully interleaved.

Using the information obtained by the profiler, OCR postcorrection in the system is made adaptive to the individual input document being processed. The adaptation affects many functionalities of the tool, e.g. by improving the detection of “suspicious” tokens (tokens of the text that are likely to present OCR errors) and the ranking of correction candidates for erroneous tokens. Both features help users to finding and correcting errors faster and in a more convenient way. Furthermore, knowledge on characteristic OCR error classes of the given input document gives a basis for the design of advanced batch processing modes during interactive correction where all errors of a particular type are inspected and corrected all at once. This feature is specifically useful when processing large documents, such as complete books or newspaper issues.

In what follows we describe a postcorrection experiment with two groups of human users. The aim of the experiment was to quantify the benefits of the batch processing features implemented in the system. The OCRed versions of two historical books[8] were presented in the postcorrection system.

After a brief introduction to the system, for each book, seven users spent 90 min trying to eliminate as many recognition errors from the input document as possible. One user group was given full access to all features of the system, taking advantage of the information provided by the automated profiling (see Fig. 3 for a sketch of the system in full mode). For the other group, for the sake of comparison, the system offered only the base functionality of a conventional postcorrection systems (highlighting of non-lexical tokens, related image view of the text, correction suggestions based on standard distance).

Every 10 min during the 90 min period, the temporary state of each user’s document was stored and the remaining number of OCR errors was evaluated. These data indicate how the accuracy of the document could be improved in the course of time, and to what extent our batch processing functionality helps to make more corrections in less time.

The diagrams in Fig. 4 show the speed how distinct users were able to reduce the number of recognition errors in the two test documents. The vertical axis gives the number OCR errors in the intermediate versions of the document of each user. Fast correctors can be identified by a graph going steeply down, indicating that a large amount of errors was eliminated in a short period of time.

*Document*1: In the diagram for Document 1 (see Fig. 4(a)), dotted curves in yellow, red and blue respectively represent Users 1–3, who worked in baseline mode. While Users 1 and 2 achieved comparable results (correcting on average 32 and 27 errors in ten minutes), User 3 corrected only 13 errors in 10 min on average (he afterwards committed that he was distracted by other tasks). Among the users with full access, User 6 (light blue graph) produced an outlier by spending some time to further explore the user interface instead of correcting errors.

Both Users 4 and 5 (brown and green graphs) were familiar with the tool and could obviously draw large profit from the tool’s full features (especially batch processing). User 7 (black graph) had been familiar with the processing of historical language, but had not worked with the tool before. He performs similarly to Users 4 and 5. In a 10 min period, these three users corrected 49 errors on average, and eventually more than 7% of all recognition errors could be eliminated within the 90 min.

*Document*2: Compared to Document 1, the characteristics of Document 2 are more convenient for the logics of the postcorrection system. The document is much larger, and it also contains more systematic errors which can easily be detected and corrected. However this does not help for the baseline Users 1 and 2 (dotted graphs in Fig. 4b): both users confirm the

[8] These books are not related to any of the above background resources.

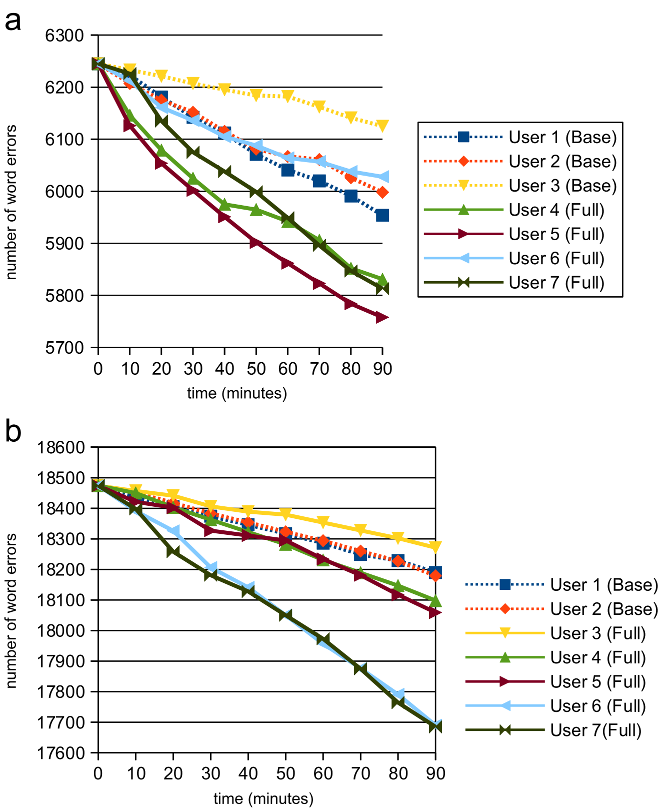


Fig. 4. Error reduction achieved by distinct users in 90 min of work. Dotted lines represent baseline users. (a) Document 1. (b) Document 2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

experience obtained from Document 1 that little more than 30 errors can be corrected in 10 min using only the manual edit functionality. Among the users with all features enabled, three groups can be distinguished. Similarly as in the first document, one user (User 3, yellow graph) performed worse than the baseline users (she reported afterwards that she had not been instructed appropriately, got confused and did not make full use of the advanced features). A second group is formed by Users 4 and 5 (green and brown graphs)—both were student assistants who had no experience with OCR and historical texts. Their average number of corrections in 10 min was 43.9—this is 36% more than the base users' average and shows clearly that also new and non-expert users can use the advanced features offered by profiling to outperform baseline users. But two experienced users (User 6 and 7, light blue and black graphs) demonstrate that much more is possible: throughout the 90-min period the batch processing modes allowed to keep up a rate of 87 corrected words in 10 minutes: this is 2.7 times faster than the baseline users and means that it took less than 7 s on average to eliminate one word error. In the period during minutes 10 and 20, User 7 actually corrected 140 words—one word every 4.3 s on average.

8. Conclusion and future work

We have described an unsupervised method that automatically analyzes OCRed historical text documents, providing a profile with information both on the specific kind of historical spelling variation and on the OCR errors found in the given input document. Documents are analyzed following a two-channel model, using a variant of the expectation–maximization method for estimating the number of recognition errors and patterns for historical spelling variation. Document profiles obtained can be exploited, e.g., for adaptive text recognition based on a feedback model, for post-correction or for improving subsequent applications such as information retrieval or information extraction.

Our evaluation results show a strong correlation between the ranks and probabilities stored in global profiles and the true distribution of OCR errors and historical patterns in the text. Experiments on the ranking of correction candidates sustained a 20% gain as compared to a baseline model using standard edit distance. Further evidence for the validity of the method has been provided by a user experiment were the benefits reached by batch corrections enabled by document specific error profiles were demonstrated and by measuring IR accessibility on the documents of the gold standard.

The profiling technology has been successfully integrated into a postcorrection system. This system, to be described in a forthcoming paper, is currently tested in several pilot projects at major European national libraries. In these pilots it is tested in more detail to which extent the postcorrection tool with underlying profiling technology can improve manual postcorrection of OCRed documents. These tests will cover at least Dutch, French, and Spanish.

In future work we would like to collaborate with industrial partners to implement a feedback mechanism based on profiling that is added to the core recognition process of an OCR engine using a specific adaptive interface. The intention is to avoid the most obvious systematic recognition errors, a major cause for distrust of the users regarding overall recognition quality. Furthermore we want to check if profiler information can be used for the detection and correction of real word errors (false friends). Earlier research has shown that a precise error model – as delivered by profiling – combined with a language model has the capability to detect these highly problematic errors.

We believe that the application domain of the profiling technology is not restricted to OCRed historical documents. Natural areas where the same technique can be applied are, e.g., OCRed modern documents, spelling correction in word processors, or query correction in search engines. In all these areas we are often confronted with systematic errors, which makes unsupervised error profiling a promising candidate for achieving improvements. In many cases, real word errors are of particular interest.

Acknowledgments

The authors were supported by EU project IMPACT. Special thanks to the Bavarian State Library, the Royal Library of the Netherlands, Austrian National Library, The French National Library and the Miguel de Cervantes Digital Library for providing access to their historical documents. Special thanks to Institute of Dutch Lexicology (INL) in Leiden for their support in language technology. Special thanks to Thorsten Vobl who implemented the graphical user interface of the postcorrection system. Special thanks to Klaus U. Schulz for advice, comments and many hours of discussions dedicated to this paper.

References

[1] G. Inc., Google books, 〈http://books.google.com/〉.
[2] R. Holley, How good can it get?, D-Lib Magazine 15(3/4).
[3] T.L. Packer, Performing information extraction to improve OCR error detection in semi-structured historical documents, in: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing (HIP '11), Beijing, China, 2011, pp. 67–74.
[4] A. Fischer, V. Frinken, A. Fornes, H. Bunke, Transcription alignment of Latin manuscripts using hidden Markov models, in: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing (HIP '11), Beijing, China, 2011, pp. 29–36.

[5] H. Balk, A. Conteh, IMPACT: centre of competence in text digitisation, in: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing (HIP '11), Beijing, China, 2011, pp. 155–160.

[6] A. Gotscharek, U. Reffle, C. Ringlstetter, K. U. Schulz, On lexical resources for digitization of historical documents, in: DocEng '09: Proceedings of the Ninth ACM symposium on Document engineering, ACM, New York, NY, USA, 2009, pp. 193–200.

[7] R. Smith, Limits on the application of frequency-based language models to ocr, in: ICDAR, IEEE, 2011, pp. 538–542.

[8] P. Yang, A. Antonacopoulos, C. Clausner, S. Pletschacher, Grid-based modelling and correction of arbitrarily warped historical document images for large-scale digitisation, in: Proceedings of the 2011 Workshop on Historical Document Imaging and Processing, HIP '11, ACM, New York, NY, USA, 2011, pp. 106–111.

[9] A. Gotscharek, U. Reffle, C. Ringlstetter, K. Schulz, A. Neumann, Towards information retrieval on historical document collections: the role of matching procedures and special lexica, International Journal of Document Analysis and Recognition (2011) 1–13.

[10] K. Taghva, T. Nartker, J. Borsack, Information access in the presence of ocr errors, in: Proceedings of the First ACM Workshop on Hardcopy Document Processing, HDP '04, ACM, New York, NY, USA, 2004, pp. 1–8.

[11] A.C. Popat, A panlingual anomalous text detector, in: Proceedings of the Ninth ACM Symposium on Document Engineering, DocEng '09, ACM, New York, NY, USA, 2009, pp. 201–204.

[12] U. Reffle, Efficiently generating correction suggestions for garbled tokens of historical language, Natural Language Engineering 17 (2011) 265–282.

[13] C.E. Shannon, A mathematical theory of communication, The Bell System Technical Journal 27 (1948) 379–423.

[14] R. Wagner, M. Fisher, The string-to-string correction problem, Journal of the ACM 21 (1) (1974) 168–173.

[15] F. Weigel, S. Baumann, J. Rohrschneider, Lexical postprocessing by heuristic search and automatic determination of the edit costs, in: Proceedings of the Third International Conference on Document Analysis and Recognition (ICDAR 95), 1995, pp. 857–860.

[16] E.S. Ristad, P.N. Yianilos, Learning string edit distance, in: Proceedings of the 14th International Conference on Machine Learning, Morgan Kaufmann, 1997, pp. 287–295.

[17] E. Brill, R.C. Moore, An improved error model for noisy channel spelling correction, in: ACL '00: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, Morristown, NJ, USA, 2000, pp. 286–293.

[18] O. Kolak, W. Byrne, P. Resnik, A generative probabilistic ocr model for nlp applications, in: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, vol. 1, NAACL '03, Association for Computational Linguistics, Stroudsburg, PA, USA, 2003, pp. 55–62.

[19] K.W. Church, W.A. Gale, Probability scoring for spelling correction, Statistics and Computing 1 (1991) 93–103.

[20] L.-M. Liu, Y.M. Babad, W. Sun, K.-K. Chan, Adaptive post-processing of ocr text via knowledge acquisition, in: Proceedings of the 19th annual conference on Computer Science, CSC '91, ACM, New York, NY, USA, 1991, pp. 558–569.

[21] S.V. Rice, G. Nagy, T.A. Nartker, Optical Character Recognition: An Illustrated Guide to the Frontier, Kluwer Academic Publishers, 1999.

[22] X. Tong, D.A. Evans, A statistical approach to automatic OCR error correction in context, in: Proceedings of the Fourth Workshop on Very Large Corpora, Copenhagen, Denmark, 1996, pp. 88–100.

[23] C. Ringlstetter, U. Reffle, A. Gotscharek, K.U. Schulz, Deriving symbol dependent edit weights for text correction—the use of error dictionaries, in: ICDAR, 2007, pp. 639–643.

[24] R. Jin, C. Zhai, A.G. Hauptmann, Information retrieval for ocr documents: a content-based probabilistic correction model, in: Document Recognition and Retrieval X Santa Clara, CA, USA, Proceedings, 2003, pp. 128–135.

[25] A. Ernst-Gerlach, N. Fuhr, Generating search term variants for text collections with historic spellings, in: Proceedings of the 28th European Conference on Information Retrieval Research (ECIR 2006), Springer, 2006, pp. 1–8.

[26] B. Jurish, More than words: Using token context to improve canonicalization of historical German, JLCL 25 (1) (2010) 23–39.

[27] K.U. Schulz, S. Mihov, Fast string correction with Levenshtein automata, International Journal of Document Analysis and Recognition 5 (1) (2002) 67–85.

Ulrich Reffle finished his PhD in Computational Linguistics in 2011. From 2007 to 2008, he was a researcher at the Center for of Information and Language Processing. Ulrich joined the EC IMPACT project on mass-digitization of historical books in 2008. Research interests are corpus linguistics, document post-processing, information retrieval in noisy environments.

Christoph Ringlstetter finished his PhD in Computational Linguistics in 2006. From 2006 to 2008, he was a postdoc at the Alberta Ingenuity Center for Machine Learning (AICML), University of Alberta, Canada. In 2008 Christoph joined EC IMPACT project on mass-digitization of historical books. Research interests are corpus linguistics, document post-processing, information retrieval in noisy environments and semantic search.