Reducing OCR errors in Gothic-script documents

Furrer, Lenz ; Volk, Martin

Abstract: In order to improve OCR quality in texts originally typeset in Gothic script, we have built an automated correction system which is highly specialized for the given text. Our approach includes external dictionary resources as well as information derived from the text itself. The focus lies on testing and improving different methods for classifying words as correct or erroneous. Also, different techniques are applied to find and rate correction candidates. In addition, we are working on a web application that enables users to read and edit the digitized text online.

Reducing OCR Errors in Gothic-Script Documents

Lenz Furrer
University of Zurich
Lenz.Furrer@uzh.ch

Martin Volk
University of Zurich
volk@cl.uzh.ch

Abstract

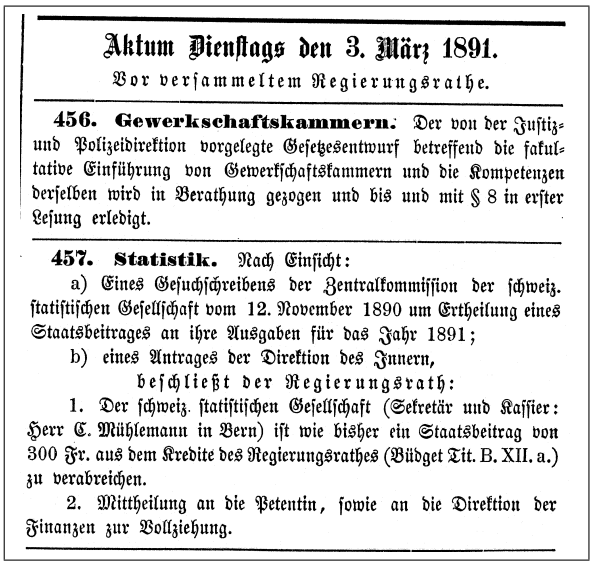
In order to improve OCR quality in texts originally typeset in Gothic script, we have built an automated correction system which is highly specialized for the given text. Our approach includes external dictionary resources as well as information derived from the text itself. The focus lies on testing and improving different methods for classifying words as correct or erroneous. Also, different techniques are applied to find and rate correction candidates. In addition, we are working on a web application that enables users to read and edit the digitized text online.

1 Introduction

The State Archive of Zurich currently runs a digitization project, aiming to make publicly available online governmental texts in German of almost 200 years. A part of this project comprises the resolutions by the Zurich Cantonal Government from 1887 to 1902, which are archived as printed volumes in the State Archive. These documents are typeset in Gothic script, also known as *blackletter* or *Fraktur*, in opposition to the *antiqua* fonts of modern typesetting. In a cooperation project of the State Archive and the Institute of Computational Linguistics at the University of Zurich, we digitize these texts and prepare them for public online access.

The tasks of the project are automatic image-to-text conversion (OCR) of the approximately 11,000 pages, the segmentation of the bulk of text into separate resolutions, the annotation of metadata (such as the date or the archival signature) as well as improving the text quality by automatically correcting OCR errors.

As for OCR, the data are most challenging, since the texts contain not only Gothic type letters – which already lead to a lower accuracy compared to antiqua texts – but also particular words, phrases and even whole paragraphs are printed in antiqua font (cf. figure 1). Although we are lucky to have an OCR engine capable of processing mixed Gothic and antiqua texts, the alternation of the two fonts still has an impairing effect on the text quality. Since the interspersed antiqua tokens can be very short (e. g. the abbreviation *Dr.*), their diverting script is sometimes not recognized by the engine. This leads to heavily misrecognized words due to the different shapes of the typefaces; for example antiqua *Landrecht* (Engl.: "citizenship") is rendered as completely illegible *I>aii<lreclitt*, which is clearly the result of using the inappropriate recognition algorithm for Gothic script.



Aktum Dienstags den 3. März 1891.

Vor versammeltem Regierungsrathe.

456. Gewerkschaftskammern. Der von der Justiz- und Polizeidirektion vorgelegte Gesetzesentwurf betreffend die fakultative Einführung von Gewerkschaftskammern und die Kompetenzen derselben wird in Berathung gezogen und bis und mit § 8 in erster Lesung erledigt.

457. Statistik. Nach Einsicht:

a) Eines Gesuchschreibens der Zentralkommission der schweiz. statistischen Gesellschaft vom 12. November 1890 um Ertheilung eines Staatsbeitrages an ihre Ausgaben für das Jahr 1891;

b) eines Antrages der Direktion des Innern,

beschließt der Regierungsrath:

1. Der schweiz. statistischen Gesellschaft (Sekretär und Kassier: Herr E. Mühlemann in Bern) ist wie bisher ein Staatsbeitrag von 300 Fr. aus dem Kredite des Regierungsrathes (Budget Tit. B. XII. a.) zu verabreichen.

2. Mittheilung an die Petentin, sowie an die Direktion der Finanzen zur Vollziehung.

Figure 1: Detail of a session header, followed by two resolutions, in German. The numbered titles as well as Roman numerals are typeset in antiqua letters.

In section 2 we present the OCR system used in the project. The focus of our work lies on section 3, which discusses our efforts in post-correcting OCR

```
- <par align="Center">
  - <line baseline="5974" l="3786" t="5870" r="5568" b="6002">
    - <formatting lang="OldGerman" ff="Arial" fs="15." spacing="-12">
        <charParams l="3786" t="5870" r="3868" b="5986" wordStart="1" wordFromDictionary="0"
          wordNormal="1" wordNumeric="0" wordIdentifier="0" charConfidence="98" serifProbability="255"
          wordPenalty="15" meanStrokeWidth="161">A</charParams>
        <charParams l="3874" t="5872" r="3926" b="5974" wordStart="0" wordFromDictionary="0"
          wordNormal="1" wordNumeric="0" wordIdentifier="0" charConfidence="92" serifProbability="255"
          wordPenalty="15" meanStrokeWidth="161">k</charParams>
        <charParams l="3932" t="5874" r="3966" b="5976" wordStart="0" wordFromDictionary="0"
          wordNormal="1" wordNumeric="0" wordIdentifier="0" charConfidence="96" serifProbability="255"
          wordPenalty="15" meanStrokeWidth="161">t</charParams>
        <charParams l="3970" t="5902" r="4022" b="5974" suspicious="1" wordStart="0"
          wordFromDictionary="0" wordNormal="1" wordNumeric="0" wordIdentifier="0" charConfidence="92"
          serifProbability="255" wordPenalty="15" meanStrokeWidth="161">u</charParams>
        <charParams l="4028" t="5900" r="4106" b="5974" wordStart="0" wordFromDictionary="0"
          wordNormal="1" wordNumeric="0" wordIdentifier="0" charConfidence="98" serifProbability="255"
          wordPenalty="15" meanStrokeWidth="161">m</charParams>
```

Figure 2: Output XML of Abbyy Recognition Server 3.0, representing the word “Aktum”.

errors. Section 4 is concerned with a web application.

2 OCR system

We use Abbyy’s Recognition Server 3.0 for OCR. Our main reason to decide in favour of this product was its capability of converting text with both antiqua and Gothic script. Another great advantage in comparison to Abbyy’s FineReader and other commercial OCR software is its detailed XML output file, which contains a lot of information about the recognized text.

Figure 2 shows an excerpt of the XML output. The fragment contains information about the five letters of the word *Aktum* (here meaning “governmental session”), which corresponds to the first word in figure 1. The first two XML tags, `par` and `line`, hold the contents and further information of a paragraph and a line respectively, e. g. we can see that this paragraph’s alignment is centered. Each character is embraced by a `charParams` tag that has a series of attributes. The graphical location and dimensions of a letter are described by the four pixel positions `l`, `t`, `r` and `b` indicating the left, top, right and bottom borders. The following boolean-value attributes specifie if a character is at a left word boundaries (`wordStart`), if the word was found in the internal dictionary (`wordFromDictionary`) and if it is a regular word or rather a number or a punctuation token (`wordNormal`, `wordNumeric`, `wordIdentifier`). `charConfidence` has a value between 0 and 100 indicating the recognition confidence for this character, while `serifProbability` rates its probability of having serifs on a scale from 0 to 255. The fourth character *u* shows an additional attribute `suspicious`, which only appears if it is set to true. It marks characters that are likely to be misrecognized (which is not the case here).

We are pleased with the rich output of the OCR system. Still, we can think of features that could make a happy user even happier. For example, post-correction systems could gain precision if the alternative recognition hypotheses made during OCR were accessible.[1] Some of the information provided is not very reliable, such as the attribute “wordFromDictionary” that indicates if a word was found in the internal dictionary: on the one hand, a lot of vocabulary is not covered, whereas on the other, there are misrecognized words that are found in the dictionary (e. g. *Bandirektion*, which is an OCR error for German *Baudirektion*, Engl.: “building ministry”). While the XML attribute “suspicious” (designating spots with low recognition confidence) can be useful, the “charConfidence” value does not help a lot with locating erroneous word tokens.

An irritating aspect is, that we found the dehyphenation to be done worse than in the previous engine for Gothic OCR (namely FineReader XIX). Words hyphenated at a line break (e. g. *fakul- tative* on the fourth line in figure 1) are often split into two halves that are no proper words, thus looking up the word parts in its dictionary does not help the OCR engine with deciding for the cor-

[1] Abbyy’s Fine Reader Software Developer Kit (SDK) is capable of this.

rect conversion of the characters. Since Abbyy's systems mark hyphenation with a special character when it is recognised, one can easily determine the recall by looking at the output. We encountered the Recognition Server's recall to be significantly lower than that of FineReader XIX, which is regrettable since it goes along with a higher error rate in hyphenated words.

The Recognition Server provides dictionaries for various languages, including so-called "Old German".[2] In a series of tests the "Old German" dictionary has shown slightly better results than the Standard German dictionary for our texts. Some, but not all of the regular spelling variations compared to modern orthography are covered by this historical dictionary, e. g. old German *Urtheil* (in contrast to modern German *Urteil*, Engl.: "jugdement") is found in Abbyy's internal dictionary, whereas *reduzirt* (modern German *reduziert*, Engl.: "reduced") is not. It is also possible to include a user-defined list of words to improve the recognition accuracy. However, when we added a list of geographical terms to the system and compared the output of before and after, there was hardly any improvement. Although the words from our list were recognized more accurately, the overall number of misrecognized words was almost completely compensated by new OCR errors unrelated to the words added.

All in all, Abbyy's Recognition Server 3.0 serves our purposes of digitizing well, especially in respect of handling large amounts of data and for pipeline processing.

3 Automated Post-Correction

The main emphasis of our project is on the post-correction of recognition errors. Our evaluation of a limited number of text samples yielded a word accuracy of 96.7 %, which means that one word out of 30 contains an error (as e. g. *Regieruug* instead of correct *Regierung*, Engl.: "government"). We aim to significantly reduce the number of misrecognized word tokens by identifying them in the OCR output and determining the originally intended word with its correct spelling.

In order to correct a maximum of errors in the corpus with the given resources, we decided for an automated approach to the post-correction. The great homogeneity of the texts concerning the layout as well as the limited variation in orthography are a good premise to achieve remarkable improvements. Having in mind the genre of our texts, we followed the idea of generating useful documents ready for reading and resigned from manually correcting or even re-keying the complete texts, as did e. g. the project *Deutsches Textarchiv*[3], which has a very literary, almost artistic standard in digitizing first-edition books from 1780 to 1900.

The task resembles that of a spell-checker as found in modern text processing applications, with two major differences. First, the scale of our text data – with approximately 11 million words we expect more than 300,000 erroneous word forms – does not allow for an interactive approach, asking a human user for feedback on every occurrence of a suspicious word form. Therefore we need an automatic system that can account for corrections with high reliability. Second, dating from the late 19th century, the texts show historic orthography, which differs in many ways from the spelling encountered in modern dictionaries (e. g. historic *Mittheilung* vs. modern *Mitteilung*, Engl.: "message"). This means that using modern dictionary resources directly cannot lead to satisfactory results.

Additionally, the governmental resolutions contain, typically, many toponymical references, i. e. geographical terms such as *Steinach* (a stream or place name) or *Schwarzenburg* (a place name). Many of these are not covered by general-vocabulary dictionaries. We also find regional peculiarities, such as pronunciation variants or words that are not common elsewhere in the German spoken area, e. g. *Hülfstrupp* vs. standard German *Hilfstruppe*, Engl.: "rescue team". Of course there is also a great amount of genre-specific vocabulary, i. e. administrative and legal jargon. We are hence challenged to build a fully-automated correction system with high precision that is aware of historic spelling and regional variants and contains geographical and governmental language.

3.1 Classification

The core task of our correction system with respect to its precision is the classification routine, that determines the correctness of every word. This part is evidently the basis for all correcting steps.

[2] The term has not to be confused with the linguistic epoch *Old High German*, which refers to texts from the first millenium A. D. Abbyy's "Old German" seems to cover some orthographical variation from the 19th, maybe the 18thcentury.

[3] see www.deutschestextarchiv.de

Misclassifications are detrimental, as they can lead to false corrections. At the same time it is also the hardest task, as there are no formal criteria that universally describe orthography. Reynaert (2008) suggests building a corpus-derived lexicon that is mainly based on word frequencies and the distribution of similarly spelled words. However, this is sometimes misleading. For example, *saumselig* (Engl.: "dilatory") is a rare but correct word, which is found only once in the whole corpus, whereas *Liegenschast* is not correct (in fact, it is misrecognized for *Liegenschaft*, Engl.: "real estate"), but it is found sixteen times in this erroneous form. This shows that the frequency of word forms alone is not a satisfactory indicator to distinguish correct words from OCR errors; other information has to be used as well.

An interesting technique for reducing OCR errors is comparing the output of two or more different OCR systems. The fact that different systems make different errors is used to localize and correct OCR errors. For example, Volk et al. (2010) achieved improvements in OCR accuracy by combining the output of Abbyy FineReader 7 and Omnipage 17. However, this approach is not applicable to our project for the simple reason that there is to our knowledge, for the time being, only one state-of-the-art OCR system that recognizes mixed antiqua / Gothic script text.

The complex problem of correcting historical documents with erroneous tokens is addressed by Reffle (2011) with a two-channel model, which integrates spelling variation and error correction into one algorithm. To cover the unpredictable number of spelling variants of texts with historical orthography (or rather, depending on their age, texts without orthography), the use of static dictionaries is not reasonable as they would require an enormous size. Luckily, the orthography of our corpus is comparatively modern and homogenous, which means that there is a manageable number of deviations in comparison to modern spelling and between different parts of the corpus.

In our approach, we use a combination of various resources for this task, such as a large dictionary system for German that covers morphological variation and compounding (such as *ging*, a past form of *gehen*, Engl.: "to go", or German *Niederdruckdampfsystem*, a compound of four segments meaning "low-pressure steam system"), a list of local toponyms (since our texts deal mostly with events in the administered area) and the recognition confidence values of the OCR engine. Every word is either judged as correct or erroneous according to the information we gather from the various resources.

The historic and regional spelling deviations are modelled with a set of handwritten rules describing the regular differences. For example, with a rule stating that the sequence *th* corresponds to *t* in modern spelling, the old form *Mittheilung* can be turned into the standard form *Mitteilung*. While the former word is not found in the dictionary, the derived one is, which allows for the assumption that *Mittheilung* is a correctly spelled word.

The classification method is applied to single words, which are thus considered as a correct or erroneous form separately and without their original context. This reduces the number of correction candidates significantly, as not every word is to be handled, but only the set of the *distinct* word forms. However, this limits the correction to non-word errors, i. e. misrecognized tokens that cannot be interpreted as a proper word. For example, *Fcuer* is clearly a non-sense word, which should be spelled *Feuer* (Engl.: "fire"). In contrast, the two word forms *Absatze* and *Absätze* (Engl.: "paragraph") are two correct morphological variants of the same word, which are often confused during OCR. To decide which one of the two variants is correct in a particular occurrence, the word has to be judged in its original context, which is outside the scope of our project.

The decision for the correctness of a word is primarily based on the output of the German morphology system Gertwol by Lingsoft, in that we check every word for its analyzability. Although the analyses of Gertwol are reliable in most cases, there are two kinds of exceptions that can lead to false classifications. First, there are correct German words that are unknown to Gertwol, such as Latin words like *recte* (Engl.: "right") or proper names. Second, sometimes Gertwol finds analyses for misrecognized words, e. g. correct *Regierungsrat* (Engl.: "member of the government") is often rendered *Negierungsrat*, which is then analysed as a compound of *Negierung* (Engl.: "negation") and *Rat* (Engl.: "councillor"). To avoid these misclassifications we apply heuristics which include word lists for known frequent issues, the frequency of the word form or the feedback values of the OCR system concerning the

recognition confidence.

3.2 Correction

The set of all words recognized as correct words plus their frequency can now serve as a lexicon for correcting erroneous word tokens. This corpus-derived lexicon is by nature highly genre-specific, which is desirable. On the other hand, rare words might remain uncorrected if all of their few occurrences happen to be misspelled, in which case there will be no correction candidate in the lexicon. Due to the repetitive character of the texts – administrative work involves many recurrent issues – there is also a lot of repetition in the vocabulary across the corpus. This increases the chance that a word misrecognized in one place has a correct occurrence in another.

The procedure of finding correction candidates is algorithmically complex, since the orthographical similarity of words is not easily described in a formal way. There is no linear ordering that would group together similar words. In the simplest approach to searching for similar words in a large list, every word has to be compared to every other word, which has quadratic complexity regarding the number of words. In our system we avoid this with two different approaches that are applied subsequently. In the first step, only a limited set of character substitutions is performed, whereas the second step allows for a broader range of deviations between an erroneous form and its correction.

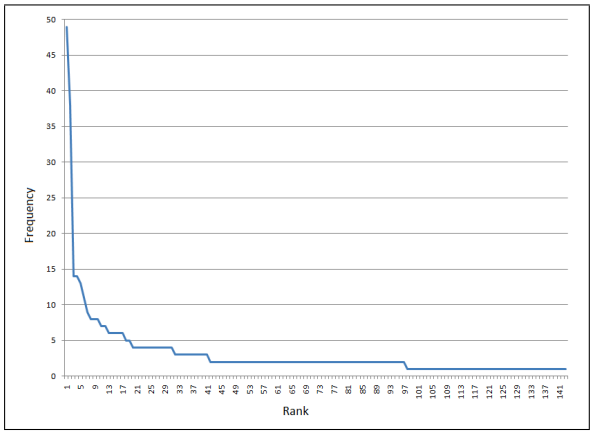


Figure 3: OCR errors ranked by their frequency, from a sample text of approx. 50,000 characters length.

3.2.1 Substitution of similar characters

In this approach we attempt to undo the most common OCR errors. When working with automatically recognized text, one will inevitably notice that a small set of errors accounts for the majority of misrecognized characters. At character level, OCR errors can be described as substitutions of single characters. Thus, by ranking these error types by their frequency, it can be shown that already a small sample of OCR errors resembles Zipfian distribution,[4] as is demonstrated in figure 3. For example, the 20 most frequent types of errors, which is less than 14 % of the 143 types encountered, sum up to 50 % of all occurrences of character errors. Table 1 shows the head of this error list as well as its tail. Among the top 6 we find pairs like *u* and *n* that are also often confounded when recognizing antiqua text as well as typical Gothic-script confusions like *d–b* or *s–f*. Figure 4 illustrates the optical similarity of certain characters that can also challenge the human eye. For example, due to its improper printing, the fifth letter cannot be determined clearly as *u* or *n*.

rank	frequency	{correct}–{recognized}
1	49	{u}–{n}
2	38	{n}–{u}
3	14	{a}–{ä}
4	14	{}–{.}
5	13	{d}–{b}
6	11	{s}–{f}
…	…	…
140	1	{o}–{p}
141	1	{r}–{?}
142	1	{§}–{8}
143	1	{ä}–{ö}

Table 1: Either ends of a ranked list, showing character errors and their frequency.

Figure 4: Gothic characters of the *Schwabacher Fraktur* that are frequently confused by the OCR system. From left to right, the characters are: *s* (in its long form), *f*, *u*, *n*, *u* or *n*, *B*, *V*, *R*, *N*.

For our correction system we use a manually edited substitution table that is based on these observations. The substitution pairs are inverted and used as replacement operations that are applied to the misspelled word forms. The spelling vari-

[4] Zipf's Law states that, given a large corpus of natural language data, the rank of each word is inversely proportional to its frequency.

ants produced are then searched for correct words that can be found in our dictionary. For example, given an erroneous token *sprechcn* and a substitution pair *e, c* stating that recognized *c* can represent actual *e*, the system produces the variants *spreehcn*, *sprechen* and *spreehen*. Consulting the dictionary finally uncovers *sprechen* (Engl.: "to speak") as a correct word, suggesting it as a correction for garbled *sprechcn*.

3.2.2 Searching shared trigrams

In a second step, we look for similar words more generally. This is applied to the erroneous words that could not be corrected by the method described above. There are various techniques to efficiently find similar words in large mounds of data, such as Reynaert's (2005) anagram hashing method, that treats words as multisets of words and models edit operations as arithmetic functions. Mihov and Schulz (2004) combine finite-state automata with filtering methods that include a-tergo dictionaries.

In our approach, we use character n-grams to find similar correct words for each of the erroneous words. Since many of the corrections with a close editing distance have already been found in the previous step, we have to look for corrections in a wider spectrum now. In order to reduce the search space, we create an index of trigrams that points to the lexicon entries. Every word in the lexicon is split into overlapping sequences of three characters, which are then stored in a hashed data structure. For example, from *ersten* (Engl.: "first") we get the four trigrams *ers*, *rst*, *ste* and *ten*. Each trigram is used as a key that points to a list of all words containing this three-letter sequence, so *ten* not only leads to *ersten*, but also to *warten* (Engl.: "wait") and many other words with the substring *ten*.

Likewise, all trigrams are derived from each erroneous word form. All lexicon entries that share at least one trigram with the word in question can thus be accessed by the trigram index. The lexicon entries found are now ranked by the number of trigrams shared with the error word. To stay with the previous example, a misrecognized word form *rrsten*[5] has three trigrams (*rst*, *ste*, *ten*) in common with *ersten*, but only one trigram (*ten*) is shared with *warten*.

The correction candidates found with this method are further challenged, e. g. by using the Levenshtein distance as a similarity measure and by rejecting corrections that affect the ending of a word.[6]

3.3 Evaluation

In order to measure the effect of the correction system, we manually corrected a random sample of 100 resolutions (approximately 35,000 word tokens). Using the ISRI OCR-Evaluation Frontiers Toolkit (Rice, 1996), we determined the recognition accuracy before and after the correction step. As can be seen in table 2, the text quality in terms of word accuracy could be improved from 96.72 % to 98.36 %, which means that the total number of misrecognized words was reduced by half.

4 Crowd Correction

Inspired by the Australian Newspaper Digitisation Program (Holley, 2009) and their online crowd-correction system, we are setting up an online application. We are working on a tool that allows for browsing through the corpus and reading the text in a synoptical view of both the recognized plain text and a scan image of the original printed page. Our online readers will have the possibility to immediately correct errors in the recognized text by clicking a word token for editing. The original word in the scan image is highlighted for convenience of the user, which permits fast comparison of image and text, so the intended spelling of a word can be verified quickly.

Crowd correction combines easy access of the historic documents with manual correcting. Similar to the concept of publicly maintained services as Wikipedia and its followers, information is provided free of charge. At the same time, the user has the possibility to give something back by im-

[5] The correction pair *ersten–rrsten* is not found in the first step although its editing distance is 1 and thus minimal. But since the error {e}–{r} has a low frequency, it is not contained in the substitution table, which makes this correction invisible for the common-error routine.

[6] Editing operations affecting a word's ending are a delicate issue in an inflecting language like German, since it is likely that we are dealing with morphological variation instead of an OCR error. Data sparseness of rare words can lead to unfortunate constellations. For example, the adjective form *innere* (Engl.: "inner") is present in the lexicon, while the inflectional variant *innern* is lacking. However, the latter form is indeed present in the corpus, but only in the misrecognized form *iunern*. As *innere* is the only valuable correction candidate in this situation, the option would be changing misspelled *iunern* into grammatically inapt *innere*. For the sake of legibility, an unorthographical word is preferred to an ungrammatical form.

	plain	(errors)	corrected	(errors)	improvement
character accuracy	99.09 %	(2428)	99.39 %	(1622)	33.2 %
word accuracy	96.72 %	(1165)	98.36 %	(584)	49.9 %

Table 2: Evaluation results of the automated correction system.

proving the quality of the corpus. It is important to keep the technical barrier low, so that new users can start correcting straight away without needing to learn a new formalism. With our click-and-type tool the access is easy and intuitive. However, in consequence, the scope of the editing operations is limited to word level, it does not, for example, allow for rewriting whole passages.

5 Conclusion

With this project we demonstrate how to build a highly specialized correction system for a specific text collection. We are using both general and specific resources, while the approach as a whole is widely generalizable. Of course, working with older texts with less standardized orthography than late 19thcentury governmental resolutions may lead to a lower improvement, but the techniques we applied are not bound to a certain epoch. In the crowd correction approach we see the possibility to further improve OCR output in combination with online access of the historic texts.

References

Rose Holley. 2009. *How Good Can It Get? Analysing and Improving OCR Accuracy in Large Scale Historic Newspaper Digitisation Programs*. D-Lib Magazine, 15(3/4).

Stoyan Mihov and Klaus U. Schulz. 2004. *Fast Approximate Search in Large Dictionaries*. Computational Linguistics 30(4):451–477.

Ulrich Reffle. 2011. *Efficiently generating correction suggestions for garbled tokens of historical language*. Natural Language Engineering 17(2):265–282.

Martin Reynaert. 2005. *Text-Induced Spelling Correction*. Ph. D. thesis, Tilburg Universtiy, Tilburg, NL.

Martin Reynaert. 2008. *Non-Interactive OCR Post-Correction for Giga-Scale Digitization Projects*. Proceedings of the Computational Linguistics and Intelligent Text Processing 9th International Conference, CICLing 2008 Berlin, 617–630.

Stephen V. Rice. 1996. *Measuring the Accuracy of Page-Reading Systems*. Ph. D. dissertation, University of Nevada, Las Vegas, NV.

Martin Volk, Torsten Marek, and Rico Sennrich. 2010. *Reducing OCR Errors by Combining Two OCR Systems*. ECAI 2010 Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH 2010), Lisbon, Portugal, 16 August 2010 – 16 August 2010, 61–65.