Published in Lecture Notes in Computer Science 7097, 103-114, 2011
which should be used for any reference to this work

Information Retrieval Strategies for Digitized Handwritten Medieval Documents

Nada Naji and Jacques Savoy

University of Neuchatel, Computer Science Department, Rue Emile-Argand 11,
2000 Neuchatel, Switzerland
{Nada.Naji,Jacques.Savoy}@unine.ch

Abstract. This paper describes and evaluates different IR models and search strategies for digitized manuscripts. Written during the thirteenth century, these manuscripts were digitized using an imperfect recognition system with a word error rate of around 6%. Having access to the internal representation during the recognition stage, we were able to produce four automatic transcriptions, each introducing some form of spelling correction as an attempt to improve the retrieval effectiveness. We evaluated the retrieval effectiveness for each of these versions using three text representations combined with five IR models, three stemming strategies and two query formulations. We employed a manually-transcribed error-free version to define the ground-truth. Based on our experiments, we conclude that taking account of the single best recognition word or all possible top-*k* recognition alternatives does not provide the best performance. Selecting all possible words each having a log-likelihood close to the best alternative yields the best text surrogate. Within this representation, different retrieval strategies tend to produce similar performance levels.

Keywords: Medieval manuscripts, IR with noisy text, OCR, handwritten text IR, Middle High German, text recognition, digital libraries.

1 Introduction

During the last decade, there has been a growing interest in building large digital libraries with the largest projects receiving national (e.g., Gallica) or international support (The European Library, or Europeana). The main motivation behind such projects is the preservation of our cultural heritage and allowing a worldwide user-friendly access to this valuable material. From a technical perspective, handling old historical documents and in particular medieval manuscripts represents a difficult task. During the image processing and text recognition phases, we faced with the artifacts surrounding the handwritten text, ink bleeding, holes and stitches on parchments, etc. Our main objective is however to perform effective searches on the transcriptions generated from these phases. Unfortunately, it is almost impossible to obtain a perfect digital transcription of the original documents, meaning that we must accept the fact that recognition errors will always reside. The level of the error rate depends on various factors such as the accuracy of the recognition system, the quality

of the contrast between the background and the ink, the regularity of the handwriting, etc., keeping in mind that a high error rate may result in low retrieval effectiveness.

The medieval manuscripts were written in a non-standardized spelling, this aspect introduced an additional challenge. By inspecting some passages we can easily find different spellings referring to the same entity. This issue will reduce the retrieval effectiveness. Moreover, the grammar used in medieval languages was clearly different once compared to that of our modern days, thus allowing more flexibility to the writer, varying from one region to another, or even from one writer to another residing in the same region. In this context, our research group is participating in the HisDoc[1] project wherein a large set of medieval handwritten manuscripts has been carefully digitized and automatically transcribed (work done at the University of Fribourg and the University of Bern respectively).

The rest of this paper is structured as follows. Section 2 provides an overview of related work while Section 3 describes the corpora used in our experiments. Section 4 outlines the indexing strategies and describes the selected IR models. Section 5 evaluates and analyzes the results obtained from applying these IR strategies.

2 Related Work

Performing effective retrieval on historical manuscripts is still an unsolved issue despite the increasing need of museums, libraries, and even the general public for easy access to historical manuscripts [1], [2]. Many studies and experiments, in both commercial and academic frameworks, have tackled the task of retrieving noisy text.

The first challenge is having to deal with the effectiveness loss caused by imperfect character recognition [3]. In this context, TREC-5 (confusion track) constitutes a useful starting point [4]. During this evaluation campaign, three different versions of a corpus written in English were made available. The first and clean version of the corpus forms the baseline, the second and third versions are the output of scanning the printed corpus having character error rates of around 5% and 20% respectively. To measure the retrieval effectiveness with the presence of noisy text, the TREC evaluation campaign chose the MRR (mean reciprocal rank) metric, which is based on the inverse of the rank of the first relevant item retrieved [5]. Such a measure reflects the concern of users wishing to find one or a few good responses to any given request.

Using this measure, the best system in TREC-5 [6] had an MRR of 0.7353 for the clean corpus, and an MRR of 0.5737 (relative difference of -22%) when facing with an error rate of 5%. For the corpus having a 20% character error rate, the MRR value was as low as 0.4978 (-32%). Similar degradation levels were obtained by other participants [4]. However, these results should be moderated, given that Tagva *et al.* [7] had shown that when using high-definition images and a high quality OCR system, the error rate could be limited to around 2%.

Dealing with medieval handwritten manuscripts instead of typed text as well as colored paper or writing media with stains, holes and stitches instead of a

[1] http://hisdoc.unine.ch

high-contrast black-and-white would certainly generate an error rate higher than the estimated 2%. Previous studies [1], [2], [3], [4] were also limited to the English language, with, basically, documents dating to the last decades of the eighteenth century (e.g. George Washington manuscripts [1], [2]). Working with older languages means that we need to face with both spelling and grammar variations. In such cases, various approaches have been suggested to deal with spelling variability as for example the plays and poems in Early Modern English (1580 – 1640) [8], [9]. Shakespeare for instance had his name spelled as "Shakper", "Shakspe", "Shaksper", "Shakspere" or "Shakspeare", but never as the current spelling. The German language is known for its compound construction (e.g., worldwide, handgun). For instance, the word *Kühlschrank* (refrigerator) is made up of two words, namely *kühl* (cold) and *Schrank* (cupboard). According to CLEF evaluation campaigns [10], splitting compound words has shown to be effective for IR purposes as the same concept can be expressed using different forms (e.g., *Computersicherheit* vs. *Sicherheit mit Computern*). It is worth mentioning that compounding was not used as frequently in Middle High German as it is in modern German. The percentage of compound nouns to nouns in the first half of the thirteenth century was around 6.8%, this ratio increased over the centuries reaching 25.2% in the modern German language [11].

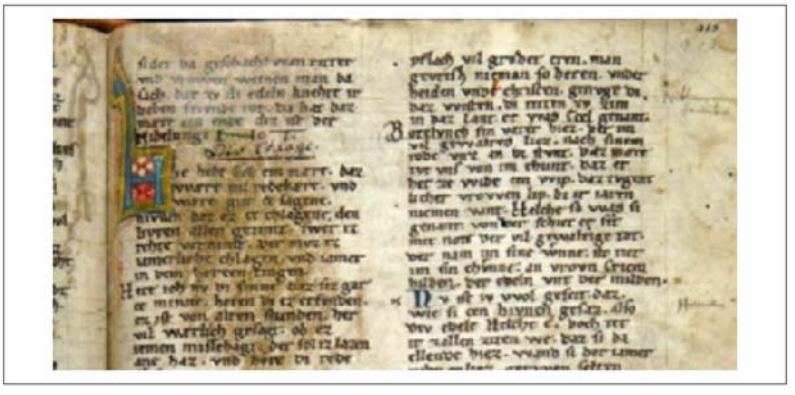

Fig. 1. A small excerpt of the Parzival manuscript

3 Evaluation Corpora and Methodology

The corpus used in our experiments is based on a well-known medieval epic poem called Parzival and is attributed to Wolfram von Eschenbach. The first version dates to the first quarter of the thirteenth century and was written in the Middle High German language. Currently, we can find several versions (with variations) but the St. Gall collegiate library *cod.* 857 is the one used for experimental evaluation [12]. An excerpt is shown in Figure 1. An error-free transcription of the poem was created manually and made available by experts. This version forms our ground-truth (GT) text and was used to assess the performance levels in our experiments.

3.1 Handwritten Recognition

In HisDoc, the manuscripts have been transcribed to the digital format using a Hidden Markov Model whose basic features are described in [12]. This recognition system is based on the closed vocabulary assumption implying that each word in the test set is known or has already appeared in the training set. This recognition system has evolved in terms of performance and achieved a word-accuracy close to 94%. Thus, the produced transcription has a word-error rate of around 6% and represents the noisy version of Parzival employed in our IR experiments. Each image corresponding to a whole page has been automatically subdivided into smaller images, each representing a single line (verse) of the poem. The line images are then processed for recognition during which the system determines, for each word, the best set of possible recognitions. Instead of being limited to a single candidate, the recognition system provides a set of seven possible words. Within each set, the seven alternatives are graded and sorted in terms of their likelihood to be correct. Thus, for each word, the seven resulting recognition#likelihood pairs are stored as [w_1#L_1, w_2#L_2, …, w_7#L_7] where w_1 is the most likely word with L_1 having the highest log-likelihood value. As a concrete example, we can inspect the word *"man"* in the verse *"dem man dirre aventivre giht"* for which the recognition pairs are: ["*man*"#36006.7, "*min*"#35656.8, "*mat*"#35452.5, "*nam*"#35424.7, "*arm*"#35296.2, "*nimt*"#35278.2, "*gan*"#35265.7]. In this case, the system succeeded to recognize this occurrence of the word *"man"* correctly as it appeared in the first position of the recognition set.

3.2 The Generation of Various Evaluation Corpora

Following medievalists' tradition, each verse (line) of the poem represents a document. The current version used in our IR experiments contains 1,328 documents corresponding to only a subset of the complete Parzival transcription. The remaining 4,477 verses form the training set used during the recognition phase and thus are not included in the search evaluation. The number of tokens per verse ranges between 2 and 9 with a mean length equal to 5.3 words.

Having access to the internal representation of the recognition system, we can investigate the quality of different output formats from an IR perspective. When facing with a word error rate of around 6%, the simplest solution is to consider only the most likely recognition for each word (e.g., using only *"man"* in our example). We will refer to this version as BW1, which represents the classical output of a recognition system. Considering that each document (verse) is quite short, the existence of a recognition error will make the retrieval of the corresponding verse an unattainable task without some sort of spelling correction or soft matching between the search keywords and the text representation. Therefore, we generated three additional corpora denoted BW3, BW7 and BWδ. The BW3 version is similar to BW1 except that the best three possible alternatives for each word were automatically included. For the same example above, the first token of the verse will be represented by the three words "*man*", "*min*" and "*mat*". Following the same vein, we generated

BW7 by incorporating all of the seven alternatives for each word. This version corresponds to the highest intensity of spelling (and recognition) variants.

Finally, we produced BWδ with a wiser strategy for incorporating term substitutes. Alternatives are included as long as the difference between the candidate's log-likelihood value and that of the most likely term is less than or equal to δ (where δ = 1.5% in the current study). Using our previous example, only the alternative "*min*" is present in addition to the term "*man*" in the BWδ version.

The benefit sought from incorporating more than one recognition alternative is to overcome the word recognition errors as well as the non-normalized spelling. For instance, the term "*Parzival*" appeared in the original manuscript as "*Parcifal*", "*Parcival*" and "*Parzifal*". All of these variants are possible and must be considered as correct spellings. Another example of spelling variants would be "*vogel*" vs. "*fogel*" (bird) and "*fisch*" vs. "*visch*" (fish) - unlike modern German, capitalizing nouns' initials was not used at that era. During the recognition phase, some of these variants may appear in the recognitions' list and using the BW3, BW7 or even BWδ corpora, some or all of them can be retained in the text representation. At this lexical level, one should also consider the inflectional morphology where various suffixes were possibly added to nouns, adjectives and names to indicate their grammatical case (e.g. as in Latin and Russian). With this additional aspect, the proper name "*Parcival*" may appear as "*Parcivale*", "*Parcivals*" or "*Parcivalen*", increasing the number of potential correct spelling variants.

3.3 Known-Item Query Generation

The manual construction of user queries together with their relevance judgments is a very costly task. A cheaper alternative is to generate them automatically. This issue had been the subject of many studies in order to obtain comparable quality between the automatically generated queries and those built by real users [13], [14], [15]. In the context of simulated query building and known-item search, we have adopted the approach suggested by [13]. This approach is based on a probabilistic framework (see Table 1) simulating the behavior of a user who wants to retrieve a known document, trying to aggregate terms that s/he recollects from the target item.

Table 1. The basic known-item query generation algorithm according to [13]

Initialize an empty query Q = { }
Select the document d_k to be the known-item with probability Prob[d_k]
Select the query length s with probability Prob[s]
Repeat s times {
Select a term t_i from the document model of d_k with probability Prob[t_i|θ_d].
Add t_i to the query Q. }
Record d_k and Q to define the (known-item / query) pair.

In adopting this algorithm, we excluded all short words (whose length is less than four characters) from being potential search terms. Moreover, words belonging to the

list of the 150 most frequent terms in the corpus were eliminated automatically. Finally, we generated three sets of 60 queries each, these are denoted QT1, QT2 and QT3 and contain single-, 2- and 3-term queries respectively.

To define the probability of selecting a given document (Prob[d_k]), we used a uniform distribution. For choosing the query terms (Prob[$t_i|\theta_d$]), each word has a chance proportional to its length (in characters), the higher the length, the higher the probability of being selected. This random process was used to generate the QT1 set.

For longer queries, we decided to augment the source of the search keywords. The underlying language model would consider the verse defining the known-item itself, and possibly the preceding and the following lines. In this random process, the verses were not assigned the same probability since the words occurring in the central verse were given twice the chance to be selected. As a final modification, when generating the QT3 set, the third query term had the possibility to originate from the short words list or the 150 frequent terms. This makes it possible to obtain nominal and prepositional phrases by having two informative terms combined with a preposition.

4 Indexing Strategies and Retrieval Models

In this paper we present a broad view of the performance achieved by various combinations of document representations and retrieval models [16]. As a first representation of verses and queries we adopted the word-based model without any stemming normalization. We used the same representation again but removed the four most common forms, namely: *der, daz, ir* and *er.* Another possible variation would be to remove all common words (e.g., the 150 most frequent forms).

As a second approach, we applied a light stemmer especially adapted for the Middle High German language. Following the principles used in an English light stemmer [17], this solution will remove a limited number of suffixes (namely "-e", "-en", and "-er") under the constraint that the resulting stem is longer than three characters. We also implemented a more aggressive stemmer to remove a larger set of both inflectional and derivational suffixes. We expect a higher effectiveness level using this approach since the best performing run in TREC-5 confusion track [6] implemented an aggressive stemmer (Porter).

Alternatively, text can be represented by overlapping sequences of n letters [18]. When setting $n = 4$ for example, the word "computing" generates the indexing terms "comp," "ompu," … and "ting." When adopting this representation strategy, the stemming procedure can thus be ignored, letting the weighting scheme assign low weights to the most frequent sequences (e.g., "ting," or "ably"). As an alternative we suggest considering only the first n letters of each word (trunc-n). When specifying $n = 4$, the word "computing" would generate the single indexing term "comp".

The terms extracted from a document can then be weighted using the classical *tf idf* formula [16]. In this case, we account for the term frequency tf_{ij} of a term t_j in a document d_i and its document frequency df_j. More precisely, we define the *idf* component as $idf_j = \log(n/df_j)$ with n indicating the number of documents in the corpus. Cosine normalization can then be applied to obtain better performance levels.

Several variants of this vector-space model have been suggested, given that the occurrence of a new term must be regarded as a rare event. In this case the first

occurrence of certain term should be given more importance than its following occurences, thus the *tf* component can be evaluated as log(*tf*)+1.

Moreover, to account for document length differences, Buckley *et al*. [19] suggest using the *Lnu-ltu* weighting method where *Lnu* and *ltu* correspond to the term weighting of the document and the query respectively (Eq. 1). This scheme was used in the best performing TREC-5 system [6].

$$w_{ij} = \frac{\left(\left(\ln(tf_{ij})+1\right) \Big/ \left(\ln(\text{mean } tf)+1\right)\right)}{(1-slope)\cdot pivot + slope \cdot nt_i} \qquad w_{qj} = \frac{\left(\ln(tf_{qj})+1\right)\cdot idf_j}{(1-slope)\cdot pivot + slope \cdot nt_i} \tag{1}$$

where nt_i indicates the number of terms in a document d_i, *pivot* and *slope* are two constants used to normalize the weights as a function of the mean document length.

As a complement to these two vector-space models, we considered two IR probabilistic schemes. First, we used the Okapi approach [20] based on the following formulation for the term t_j in document d_i.

$$w_{ij} = [(k_1+1)\cdot tf_{ij}] \,/\, (\mathrm{K} + tf_{ij}) \quad \text{where } \mathrm{K} = k_1 \cdot [(1\text{-}b) + ((b \cdot l_i) \,/\, mean\ dl)] \tag{2}$$

where l_i is the length of the d_i document, b, k_1 and *mean dl* are constants whose values are set to 0.55, 1.2 and 5.3 respectively. The second probabilistic scheme is the I(n_e)B2 model, a member of the *Divergence from Randomness* (DFR) family [21]. In this case, the weight w_{ij} reflecting the weight of term t_j in document d_i was a combination of two information measures as follows:

$$\begin{aligned} &w_{ij} = \mathrm{Inf}^1_{ij} \cdot \mathrm{Inf}^2_{ij} = \mathrm{Inf}^1_{ij} \cdot (1 - \mathrm{Prob}^2_{ij}) \quad \text{and} \\ &\mathrm{Prob}^2_{ij} = 1 - [(tc_j+1) \,/\, (df_j \cdot (tfn_{ij}+1))] \\ &\mathrm{Inf}^1_{ij} = tfn_{ij} \cdot \log_2[(n+1) \,/\, (n_e+0{,}5)] \quad \text{with } n_e = n \cdot [1 - [(n\text{-}1)/n]^{tc_j}] \end{aligned} \tag{3}$$

where tc_i represents the collection frequency of the term t_j.

Finally, we used the language model (LM) [22] in which the probability estimates were based directly on the occurrence frequencies in a document d_i, or in the corpus C. In this paper, we chose to implement Hiemstra's model [22] (Eq. 4), combining an estimate based on a document (Prob[t_j|d_i]) and a corpus (Prob[t_j|C]).

$$\mathrm{Prob}[d_i \mid q] = \mathrm{Prob}[d_i] \,.\, \textstyle\prod_{tj \in Q} [\lambda_j \,.\, \mathrm{Prob}[t_j \mid d_i] + (1\text{-}\lambda_j) \,.\, \mathrm{Prob}[t_j \mid C]] \tag{4}$$

$$\mathrm{Prob}[t_j \mid d_i] = tf_{ij} \,/\, nt_i \quad \text{and} \quad \mathrm{Prob}[t_j \mid C] = df_j \,/\, lc \quad \text{with } lc = \textstyle\sum_k df_k \tag{5}$$

In this formula, λ_j is a smoothing factor (set to a constant value equal to 0.35 for all terms t_j), and *lc* is an estimation of the size of the corpus C.

5 Evaluation

As a retrieval effectiveness measure, the TREC-5 evaluation campaign selected the MRR (mean reciprocal rank) approach which is based on the inverse of the rank of the first relevant item retrieved [4], [5]. Such a measure reflects the user concern

wishing to find one or a few good responses to a given request. In our case, we can apply the same evaluation methodology. As the relevant items are verses, returning the verse immediately before or after the correct one should not be regarded as fully-impertinent, particularly when knowing that the user usually reads a passage (a few verses) instead of a single line. Considering the immediate neighbor verses to be fully-relevant is inadequate either. To allow some degree of relevance for adjacent lines, we need to adapt the strict MRR computation to support graded relevance.

In this vein, Eguchi *et al.* [23] proposed a metric for the task of finding "one highly relevant document" called *Weighted Reciprocal Rank* (WRR), a solution improved in [24]. The word "highly" obviously implies some sort of graded relevance, but this metric imposes that a "*partially* relevant document at rank 1 is more important than ranking a *highly* relevant document at rank 2."

In this context, there exist three possible cases (where the first two are identical to the classical MRR scheme). When the search is unsuccessful; the query is evaluated as 0, while if the rank of the fully relevant document is better than these of its neighbors, the query is then evaluated as 1/R (the reciprocal of the rank of the relevant item). Third, when one of the two neighbors appears in a better rank than that of the target item (ranks denoted as R_n and R_t respectively), the query is evaluated as $\text{Max}[1/R_t; \gamma 1/R_n]$. With γ=0.5, suppose having R_n=2 and R_t = 3, thus Max[1/3; 0.5·1/2] = 1/3. In this example, the rank of the relevant item is not really far from its neighbor, the evaluation is based on the former. Having R_n = 2 and R_t = 5, the query evaluation is then Max[1/5; 0.5·1/2] = 0.25. In this case, the evaluation depends on the rank of R_n as R_t is too late to be favored to R_n. We will refer to this evaluation scheme as G(M)RR for *Graded (Mean) Reciprocal Rank*.

5.1 Evaluation of the Recognition Corpora

Since our evaluation measure focuses on high precision, we deemed a word-based representation without any morphological normalization (performance shown under the label "No stem" in Table 2) would be adequate. Alternatively, a light stemmer should also provide comparable or even better performance levels since it simply removes a very limited number of plural suffixes (labelled "Light"). Finally, the evaluation when eliminating some derivational suffixes, as it is the norm in many IR empirical studies, is shown under the label "Aggressive" in Table 2. As an alternative to the word-based model, we selected the *n*-gram model with a value of $n = 4$ which usually provides good performance levels. Another option was to apply the trunc-*n* scheme with $n = 4$, a strategy found effective for different corpora [18].

Concerning the IR models, we selected two vector-space schemes (*Lnu-ltu* and *tf idf*), two probabilistic models (Okapi, DFR-I(n_e)B2), and a language model (LM). Based on other experiments, we selected the BWδ as the source for building the text representation since this corpus has generally demonstrated the best performance. We also implemented an approach to automatically decompose compound terms which resulted in a slight improvement. The performance values shown in Table 2 & Table 3 and Table 4 were obtained using the 3-term and single-term queries respectively. Table 2 reports results from experiments based on the BWδ corpus. As can be seen, the Okapi model yields the best results regardless of the stemmer (columns "No stem", "Light" or "Aggressive") or representation used. As indicated in the last line of

Table 2, the mean performance differences between the various text representations are rather small, varying from -1.1% (4-gram) to -2.34% (trunc-4) when compared to an approach ignoring stemming normalization. In this table, statistically significant differences in performance based on the *t*-test (significance level $\alpha = 5\%$) compared to the best approaches (depicted in bold) are marked with an asterisk (*). As shown, the performance differences between the Okapi and the LM models are rather small and non-significant. However, with the *tf idf* model, the performance differences are usually statistically significant. It can be seen from Table 3 that BW1 is the best performing corpus followed by BWδ where the baseline is the error-free ground-truth corpus (GT) with differences in performance equal to -1.10% and -4.19% respectively. The performance differences between BW1 and GT are always non-significant. On the other hand, BWδ provides better retrieval effectiveness than BW1 for the QT1 set. This leads us to the conclusion that BWδ and BW1 represent the best text surrogates and that BWδ outperforms BW1 when considering single-term queries. From the MRR values shown in Table 4 for the single-term queries (QT1) and the various corpora, we can also conclude that BW3 and BW7 do not constitute pertinent alternatives. In Tables 3 and 4, we have, once again, applied the *t*-test (significance level $\alpha = 5\%$) using the performance achieved by the GT as a baseline. Significant performance differences compared to the GT levels are denoted with an asterisk (*). As shown in these tables, the performance differences between the GT and the BW3 and BW7 corpora are usually statistically significant.

Table 2. GMRR for the BWδ corpus with five IR models, using three stemming strategies, 4-gram, and trunc-4 representations, with the QT3 (60 queries)

Representation	word-based			4-gram	trunc-4
IR Model	No stem	Light	Aggressive		
Okapi	**0.6706**	**0.6586**	**0.6528**	0.6471	**0.6503**
DFR-I(n_e)B2	0.6161*	0.6145	0.6183	0.6473	0.6336
LM (λ=0,35)	0.6647	0.6485	0.6466	**0.6504**	0.6479
Lnu-ltu	0.6544	0.6315	0.6379	0.6393	0.6240
tf idf	0.6313	0.6146*	0.6193	0.6176*	0.6057*
Difference %		-2.15%	-1.92%	-1.10%	-2.34%

Table 3. GMRR for different recognition corpora together with the ground-truth (GT) using five IR models, aggressive stemmer (QT3, 60 queries)

IR Model	GT	BWδ	BW1	BW3	BW7
Okapi	0.6594	**0.6528**	0.6555	0.6406	0.5866*
DFR-I(n_e)B2	0.6572	0.6183	0.6528	0.6358	0.5808*
LM (λ=0,35)	0.6642	0.6466*	**0.6596**	**0.6491**	**0.5875***
Lnu-ltu	0.6649	0.6379	**0.6596**	0.6130	0.5866*
tf idf	**0.6679**	0.6193*	0.6495	0.5956*	0.5405*
Difference %		-4.19%	-1.10%	-5.42%	-13.03%

Table 4. MRR for different recognition corpora together with the ground-truth (GT) using five IR models, aggressive stemmer (QT1, 60 queries)

IR Model	GT	BWδ	BW1	BW3	BW7
Okapi	0.6123	0.6030	0.5839	0.4001*	0.3184*
DFR-I(n_e)B2	0.6123	0.5950	0.5839	0.4001*	0.3184*
LM (λ=0,35)	0.6123	0.5951	0.5839	0.4078*	0.3085*
Lnu-ltu	0.6121	0.6210	0.5837	**0.4100***	0.3218*
tf idf	**0.6271**	**0.6365**	**0.5890**	0.3956*	**0.3271***
Difference %		-0.83%	-4.94%	-34.54%	-48.18%

5.2 Selected Query-by-Query Analyses

Single-term Queries #4, #11, #25, and #43 are composed of a rare term each, having, according to the GT corpus, *df* values equal to 1 (term appearing only in the known-item). Using the BW1 corpus and Okapi model, none of these four queries got any pertinent items retrieved. Query #4 "*machete*", for instance, has the verse "*er machete ê daz er gein ir sp(ra)ch*" as its known-item. Using the BW1 corpus, the most likely recognition of the term "*machete*" is "*machen*" which is an incorrect recognition. Since this is the only occurrence of the term in the collection, the system failed to retrieve any documents in that case. Using the BWδ corpus, the known-item was retrieved at Rank 1 since the term *"machete"* was the second likely recognition with a log-likelihood difference of less than 1.5%. This second possible recognition was therefore included in the text representation. Using the BW3 or BW7 corpora, the known-item was also retrieved but in lower ranks, in Rank 3 for BW3 and 18 for BW7. This poor performance is due to the fact that BW3 and BW7 include more alternatives which merely acted as noise in this case. Non-relevant documents containing the search term alternatives are now competitors as the final ranking depends on the *tf* values. A non-relevant item yet with a higher *tf* will thus appear before the target verse in the ranked list of retrieved items. For the same examples using the classical *tf idf* vector-space model, the known-items were retrieved in all cases (BW1, BW3, BW7, and BWδ) in acceptably high rankings: 3, 2, 2 and 7 or in positions 7, 7, 2 and 15 using DFR-I(n_e)B2. Applying an automatic decompounding strategy may provide some successful improvement, particularly for longer queries (three terms). The use of either a light or a more aggressive stemming approach was also usually more beneficial for longer queries.

6 Conclusion

In this study we investigated the underlying issues when facing with a noisy corpus having a word error rate of around 6%, originating from medieval manuscripts handwritten in Middle High German and digitized via a text recognition device. In addition to this issue, difficult matching between queries and documents can be caused by the non-standardized spelling used in medieval languages and the presence of less strict grammatical rules. Having access to the internal representation of the recognition phase, we generated the BW1 corpus by considering only the best recognition for each

input word. We also created the BW3 and BW7 corpora retaining, respectively, the best three and seven alternatives for each word. The fourth version, BWδ, includes all possible word recognition(s) having a log-likelihood less than or equal to 1.5% compared to the highest value. The error-free ground-truth transcription had been manually transcribed by the experts and served as the evaluation baseline. Based on three text representation formats (word-based, *n*-gram, trunc-*n*), five IR models (*tf idf*, *Lnu-ltu*, Okapi, DFR-I(n_e)B2, LM), three stemmers (none, light, and aggressive), and two query formulations (single-term, 3-term), we found that the best retrieval effectiveness was usually produced by the Okapi and LM models. We cannot clearly determine the best text representation as the mean differences among them are rather small. Regarding the stemming procedure, we suggest applying either an aggressive stemming that tends to produce slightly better retrieval performance when facing with longer queries. On the other hand, ignoring the stemming normalization with short queries offers usually the best performance. We have assessed the various recognition corpora against the ground-truth version. Compared to this error-free version, the BWδ shows a mean degradation in retrieval performance ranging from -4.19% (3-term queries) to -6.05% (single-term queries). When using the classical output of the recognition process (recognition output limited to a single term, or BW1), the degradation in mean performance ranges from 1.1% (3-term queries) to 10.24% (single-term queries). Considering systematically three (BW3) or seven (BW7) alternatives per input word usually tends to cause the retrieval effectiveness to decrease significantly (from -5.42% for 3-term queries to 64.34% with single-term queries).

With very few media written in older languages (newspapers did not exist in the thirteenth century), the corpus size is limited to a subset of the manuscript pages with their corresponding error-free transcriptions serving as the ground-truth text during the evaluation. Another issue that arises is that manual transcription of medieval manuscripts is a very costly task as it is quite time-consuming and can solely be performed by the experts. The best practices found and the conclusions drawn from our experiments can be applied to further manuscripts, hence making them digitally accessible and effectively searchable with the least cost possible by partially or totally eliminating the need to having them manually transcribed which will result in saving a lot of resources (time, human effort, money, etc.). With these documents being totally searchable via digital means, real user needs (queries) will thus be obtained via search requests from experts as well as public users, which in turn, will help improve the performance a great deal.

Acknowledgment. This research is supported by the Swiss NSF under Grant CRSI22_125220.

References

1. Toni, M., Manmatha, R., Lavrenko, V.: A Search Engine for Historical Manuscript Images. In: Proceedings of the ACM-SIGIR, pp. 369–376. The ACM Press, New York (2004)
2. Nicholas, R., Toni, M., Manmatha, R.: Boosted Decision Trees for Word Recognition in Handwritten Document Retrieval. In: Proceedings of the ACM-SIGIR, pp. 377–383. The ACM Press, New York (2005)
3. Callan, J., Kantor, P., Grossman, D.: Information Retrieval and OCR: From Converting Content to Grasping Meaning. SIGIR Forum 36(2), 58–61 (2002)

4. Voorhees, E.M., Garofolo, J.S.: Retrieving Noisy Text. In: Voorhees, E.M., Harman, D.K. (eds.) TREC, Experiment and Evaluation in Information Retrieval, pp. 183–197. The MIT Press, Cambridge (2005)
5. Buckley, C., Voorhees, E.: Retrieval System Evaluation. In: Voorhees, E.M., Harman, D.K. (eds.) TREC, Experiment and Evaluation in Information Retrieval, pp. 53–75. The MIT Press, Cambridge (2005)
6. Ballerini, J.P., Büchel, M., Domering, R., Knaus, D., Mateev, B., Mittendorf, E., Schäuble, P., Sheridan, P., Wechsler, M.: SPIDER Retrieval System at TREC-5. In: Proceedings of TREC-5, pp. 217–228. NIST Publication #500-238 (1997)
7. Tagva, K., Borsack, J., Condit, A.: Results of Applying Probabilistic IR to OCR Text. In: Proceedings of the ACM-SIGIR, pp. 202–211. The ACM Press, New York (1994)
8. Craig, H., Whipp, R.: Old Spellings, New Methods: Automated Procedures for Indeterminate Linguistic Data. Literary & Linguistic Computing 25(1), 37–52 (2010)
9. Pilz, T., Luther, W., Fuhr, N., Ammon, U.: Rule-Based Search in Text Databases with Nonstandard Orthography. Literacy & Linguistic Computing 21(2), 179–186 (2006)
10. Peters, C., Gonzalo, J., Braschler, M., Kluck, M. (eds.): CLEF 2003. LNCS, vol. 3237. Springer, Heidelberg (2004)
11. Gardt, A., Hauss-Zumkehr, U., Roelcke, T.: Sprachgeschichte als Kulturgeschichte. Walter de Gruyter, Berlin (1999)
12. Fischer, A., Wüthrich, M., Liwicki, M., Frinken, V., Bunke, H., Viehhauser, G., Stolz, M.: Automatic Transcription of Handwritten Medieval Documents. In: 15th International Conference on Virtual Systems and Multimedia (2007)
13. Azzopardi, L., de Rijke, M.: Automatic Construction of Known-Item Finding Test Beds. In: Proceeding ACM SIGIR, pp. 603–604. The ACM Press, New York (2006)
14. Callan, J., Connell, M.: Query-Based Sampling of Text Databases. Information Systems 19(2), 97–130 (2001)
15. Jordan, C., Watters, C., Gao, Q.: Using Controlled Query Generation to Evaluate Blind Relevance Feedback Algorithms. In: Proceedings of the Sixth ACM/IEEE-CS Joint Conference on Digital Libraries, pp. 286–295. The ACM Press, New York (2006)
16. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
17. Harman, D.: How Effective is Suffixing. Journal of the American Society for Information Science 42(1), 7–15 (1991)
18. McNamee, P., Mayfield, J.: Character *n*-gram Tokenization for European Language Text Retrieval. IR Journal 7(1-2), 73–97 (2004)
19. Buckley, C., Singhal, A., Mitra, M., Salton, G.: New Retrieval Approaches using SMART. In: Proceedings of TREC-4, pp. 25–48. NIST Publication #500-236 (1996)
20. Robertson, S.E., Walker, S., Beaulieu, M.: Experimentation as a Way of Life: Okapi at TREC. Information Processing & Management 36(1), 95–108 (2000)
21. Amati, G., van Rijsbergen, C.J.: Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness. ACM Transactions on Information Systems 20(4), 357–389 (2002)
22. Hiemstra, D.: Using Language Models for Information Retrieval. CTIT Ph.D. Thesis (2000)
23. Eguchi, K., Oyama, K., Ishida, E., Kando, N., Kuriyama, K.: Overview of the Web Retrieval Task at the Third NTCIR Workshop. NII Publication (2003)
24. Sakai, T.: Bootstrap-Based Comparisons of IR Metrics for Finding One Relevant Document. In: Ng, H.T., Leong, M.-K., Kan, M.-Y., Ji, D. (eds.) AIRS 2006. LNCS, vol. 4182, pp. 374–389. Springer, Heidelberg (2006)