Cross-Evaluation of Graph-Based Keyword Spotting in Handwritten Historical Documents

Michael Stauffer[1(✉)], Paul Maergner[2], Andreas Fischer[2,3], and Kaspar Riesen[1]

[1] Institute for Information Systems,
University of Applied Sciences and Arts Northwestern Switzerland,
Riggenbachstrasse 16, 4600 Olten, Switzerland
{michael.stauffer,kaspar.riesen}@fhnw.ch
[2] Department of Informatics, University of Fribourg,
Boulevard de Pérolles 90, 1700 Fribourg, Switzerland
{paul.maergner,andreas.fischer}@unifr.ch
[3] Institute of Complex Systems,
University of Applied Sciences and Arts Western Switzerland,
Boulevard de Pérolles 80, 1700 Fribourg, Switzerland

Abstract. In contrast to statistical representations, graphs offer some inherent advantages when it comes to handwriting representation. That is, graphs are able to adapt their size and structure to the individual handwriting and represent binary relationships that might exist within the handwriting. We observe an increasing number of graph-based keyword spotting frameworks in the last years. In general, keyword spotting allows to retrieve instances of an arbitrary query in documents. It is common practice to optimise keyword spotting frameworks for each document individually, and thus, the overall generalisability remains somehow questionable. In this paper, we focus on this question by conducting a cross-evaluation experiment on four handwritten historical documents. We observe a direct relationship between parameter settings and the actual handwriting. We also propose different ensemble strategies that allow to keep up with individually optimised systems without *a priori* knowledge of a certain manuscript. Such a system can potentially be applied to new documents without prior optimisation.

Keywords: Keyword spotting · Handwritten historical documents · Graph-based representations · Hausdorff Edit Distance · Ensemble methods

1 Introduction

Different handwritten historical documents often show large variations in the handwriting (e.g. scale or style) and are often negatively affected by ink-bleed through, fading, etc. Consequently, an automatic full transcription is often not

D. Conte et al. (Eds.): GbRPR 2019, LNCS 11510, pp. 45–55, 2019.
https://doi.org/10.1007/978-3-030-20081-7_5

feasible [3]. For this reason, *Keyword Spotting (KWS)* has been proposed as a more flexible and error-tolerant alternative [5]. In particular, KWS systems allow to retrieve all word instances in handwritten historical documents that represent a given query word.

1.1 Related Work

In graph-based KWS, a query graph is commonly matched with the graphs that represent the document words. Hence, sorted graph dissimilarities can be used to derive a retrieval index that consists – in the best case – of all relevant keywords as its top results.

Different graph-based approaches for KWS are based on different representations of the handwriting. However, nodes are often used to represent characteristic points (so called *keypoints*) in the handwriting, while edges are commonly used to represent handwriting strokes [13]. In other approaches the nodes are used to represent prototype strokes, while edges are used to connect nodes that stem from the same connected component [2,8]. More recently, a set of different graph-based handwriting representations has been proposed that make use of keypoints, grid-wise segmentations, or projection profiles [10]. These handwriting graph representations have been actually employed in various graph-based KWS applications [1,7,11,12]. Very recently, Deep Learning techniques (so called *Message Passing Neural Networks*) have been used to enhance node labels by a structural node context [7].

Regardless the graph representation actually used, a matching procedure is required in order to conduct KWS. To this end, different graph dissimilarities have been employed like, for instance, *Bipartite Graph Edit Distance (BP)* [2,8,11–13][1] as well as *Hausdorff Edit Distance (HED)* [1,7][2]. Moreover, *ensemble methods* have been proposed to combine different graph representations [11].

1.2 Contribution

It is common practice in the field of KWS research that parameters are individually optimised for every document [2,3,5,7,8,13]. That is, the parameters are often optimised on a subset of a specific document and then tested on a disjoint set stemming from the same document. However, this practice does not reflect a realistic scenario especially as libraries often keep thousands of different handwritten historical documents. It would be a very cumbersome and time consuming task to individually optimise a given KWS system for each of these documents.

In the present paper, we evaluate the generalisability of a graph-based KWS system. That is, we investigate the performance and limitation of this system in a cross-evaluation experiment on four handwritten historical documents,

[1] BP has been introduced in [9].
[2] HED has been introduced in [4].

(a) (b) (c) (d)

Fig. 1. Exemplary excerpts of four handwritten historical documents: (a) George Washington (GW), (b) Parzival (PAR), (c) Alvermann Konzilsprotokolle (AK), (d) Botany (BOT).

viz. *George Washington (GW)*[3], *Parzival (PAR)*[4], *Alvermann Konzilsprotokolle (AK)*, and *Botany (BOT)*[5]. In particular, we optimise parameters on one document (for instance GW) and eventually test the optimised settings on the three remaining documents (in this case PAR, AK, and BOT). We repeat this procedure for each document. Moreover, we propose and evaluate novel ensemble methods that allow to test unknown documents without prior optimisation step. That is, these ensemble systems combine the results of three KWS systems (individually optimised on three different manuscripts) in order to instantly perform KWS on an unseen document.

In Fig. 1, excerpts from each document are shown. The large variations in the writing styles and document states are clearly visible and illustrate the challenging task of tuning a KWS system on one document that eventually returns reasonable results on other documents.

The remainder of this paper is organised as follows. First, the graph-based KWS framework actually employed for our research study is reviewed in Sect. 2. Next, the cross-evaluation experiment on the four handwritten documents as well as the ensemble results are presented and discussed in Sect. 3. Finally, we draw conclusions and discuss further research activities in Sect. 4.

2 Graph-Based Keyword Spotting

In this section, we review a graph-based KWS framework originally proposed in [1,12]. We use this framework as basic system to conduct both the cross validation and the ensemble experiments. This framework is based on three different

[3] George Washington Papers at the Library of Congress, 1741–1799: Series 2, Letterbook 1, pp. 270–279 & 300–309, http://memory.loc.gov/ammem/gwhtml/gwseries2.html.

[4] Parzival at IAM historical document database, http://www.fki.inf.unibe.ch/databases/iam-historical-document-database/parzival-database.

[5] Alvermann Konzilsprotokolle and Botany at ICFHR2016 benchmark database, http://www.prhlt.upv.es/contests/icfhr2016-kws/data.html.

processing steps (as illustrated in Fig. 2) and is briefly outlined in the next three subsections. In the fourth and last subsection we discuss a possibility to build an ensemble out of different KWS systems that might be particularly useful in order to increase the generalisability of a KWS system.

2.1 Image Preprocessing

For the two documents GW and PAR, general noise is addressed by means of *Difference of Gaussians* filtering. Next, document images are binarised by global thresholding. Moreover, the resulting document images are automatically segmented into single word images by means of their projection profiles, and if necessary manually corrected. That is, we focus on the KWS process itself and assume perfectly segmented documents in our evaluations. For deskewing, the angle between x-axis and lower baseline of a text line is estimated and used to rotate single word images. Finally, preprocessed word images are skeletonised by means of thinning.

For the two documents AK and BOT, segmented word images are directly taken from the ICFHR2016 benchmark database [6], and thus, only binarisation has been employed. To handle small segmentation errors, we employ an additional image preprocessing step that removes small connected components on these two manuscripts.

We denote preprocessed and skeletonised word images by S from now on. For more details on the preprocessing step we refer to [11,12].

2.2 Handwriting Graphs

In general, a graph g is defined as a four-tuple $g = (V, E, \mu, \nu)$ where V and E are finite sets of nodes and edges, and $\mu : V \to L_V$ and $\nu : E \to L_E$ are labelling functions for nodes and edges, respectively. The handwriting graphs employed in this paper are defined as follows. Nodes are used to represent characteristic points, so-called *keypoints*, in the handwriting, while edges are used to represent strokes between keypoints. Hence, nodes are labelled with two-dimensional numerical labels, while edges remain unlabelled, i.e. $L_V = \mathbb{R}^2$ and $L_E = \emptyset$. In the following paragraphs we briefly review the procedure of extracting graphs from word images (for details we refer to [12]).

First, end points and junction points are identified in the word images S. Selected keypoints are added to the graph as nodes and labelled with their respective (x, y)-coordinates. Next, intermediate points are added as nodes along the skeleton in equidistant intervals of size D. Eventually, an undirected edge (u, v) between $u \in V$ and $v \in V$ is inserted into the graph for each pair of nodes that is directly connected by a chain of foreground pixels in image S.

To reduce scaling variations, the (x, y)-coordinates of the node labels $\mu(v)$ are normalised by a z-score. Formally, we replace (x, y) by $(\hat{x}, \hat{y})$, where

$$\hat{x} = \frac{x - \mu_x}{\sigma_x} \text{ and } \hat{y} = \frac{y - \mu_y}{\sigma_y}.$$

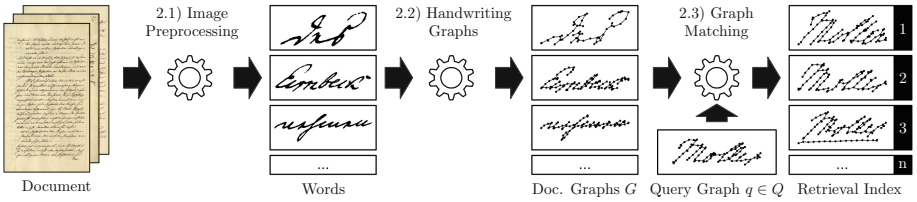

Fig. 2. Graph-based keyword spotting processing of the word "Möller".

Thereby (μ_x, μ_y) and (σ_x, σ_y) represent the mean and standard deviation of all (x, y)-coordinates in the graph under consideration[6].

For each manuscript, an original word image, a preprocessed word image, a skeletonised word image, as well as the corresponding handwriting graph is given in Fig. 3.

Fig. 3. Exemplary graph representation of four handwritten historical documents (viz. George Washington (GW), Parzival (PAR), Alvermann Konzilsprotokolle (AK), and Botany (BOT)): (a) Original word image, (b) Preprocessed word image, (d) Skeletonised word image, (c) Handwriting graph.

2.3 Graph Matching

The actual keyword spotting is based on a pairwise matching of a query graph q with all graphs g stemming from the set of document graphs G. In this paper, we make use of *Hausdorff Edit Distance (HED)* [4]. HED is a quadratic time lower bound of *Graph Edit Distance* that measures the minimum-cost deformation needed to transform one graph $g_1 = (V_1, E_1, \mu_1, \nu_1)$ into another graph $g_2 = (V_2, E_2, \mu_2, \nu_2)$ by means of deletions $(u \rightarrow \epsilon)$, insertions $(\epsilon \rightarrow v)$, and

[6] Note that the resulting graphs are available under http://www.histograph.ch/.

substitutions ($u \rightarrow v$) of nodes $u \in V_1$ and $v \in V_2$. Likewise, edit operations are defined for the edges. Formally, the HED of two graphs g_1 and g_2 can be derived by

$$\mathrm{HED}(g_1, g_2) = \sum_{u \in V_1} \min_{v \in V_2 \cup \{\epsilon\}} f(u, v) + \sum_{v \in V_2} \min_{u \in V_1 \cup \{\epsilon\}} f(u, v),$$

where $f(u, v)$ is a cost function that takes into account the node edit cost $c(u \rightarrow v)$ as well as the edge edit cost $c(q \rightarrow r)$ for all edges q and r adjacent to u and v, respectively.

The cost model employed is based on a constant cost $\tau_v \in \mathbb{R}^+$ for node deletions/insertions and a constant cost $\tau_e \in \mathbb{R}^+$ for edge deletions/insertions. For node substitutions, the following weighted Euclidean distance is employed:

$$\sqrt{\alpha \left(\sigma_x (x_i - x_j)\right)^2 + (1 - \alpha) \left(\sigma_y (y_i - y_j)\right)^2},$$

where $\alpha \in [0, 1]$ denotes a parameter to weight the importance of the x- and y-coordinate of a node, while σ_x and σ_y denote the standard deviation of all node coordinates in the current query graph q. Edge substitutions are free of cost (since they are unlabelled). We additionally use a weighting factor $\beta \in [0, 1]$ to weight the relative importance of the overall node and edge edit costs.

Finally, a retrieval index r is derived. In particular, HED is normalised by the maximum possible graph edit distance between q and g (i.e. the sum that results from deleting all nodes and edges of q and inserting all nodes and edges in g). Formally,

$$r(q, g) = \frac{\mathrm{HED}(q, g)}{\left(|V_q| + |V_g|\right) \tau_v + \left(|E_q| + |E_g|\right) \tau_e}.$$

2.4 Ensemble Methods

In order to increase the generalisability of the proposed framework, we propose three different ensemble methods that allow to combine optimised cost models of known documents. The general idea of these systems is as follows. We assume that we have three documents at hand on which a KWS system can be individually optimised. We eventually apply all three parametrisations to one unknown document and combine the three results by means of a statistical measure. Formally,

$$\begin{aligned} r_{\min}(q, g) &= \min_{i \in \{A,B,C\}} r_i(q, g), \\ r_{\max}(q, g) &= \max_{i \in \{A,B,C\}} r_i(q, g), \\ r_{\mathrm{mean}}(q, g) &= \mathrm{mean}_{i \in \{A,B,C\}} r_i(q, g), \end{aligned}$$

where $\{A, B, C\}$ represent three given manuscripts, and r_i refers to the HED optimised on manuscript A, B, or C. If we assume, for instance, that BOT is an unknown document, $\{A, B, C\}$ is given by the three remaining documents that is $A = \mathrm{GW}$, $B = \mathrm{PAR}$, and $C = \mathrm{AK}$.

3 Experimental Evaluation

3.1 Experimental Setup

For all evaluations, the accuracy is measured by the *Mean Average Precision (MAP)*, which is the mean area under all recall-precision curves of all individual keywords. In particular, the evaluation is conducted in two steps, viz. validation and test.

First, ten different keywords (with different word lengths) are manually selected on each dataset. Based on these keywords, we define an independent validation set for parameter optimisation that consists of 10 random instances per keyword instance and 900 additional random words (in total 1,000 words). We evaluate 25 pairs of constants for node and edge deletion/insertion costs ($\tau_v = \tau_e \in \{1, 4, 8, 16, 32\}$) in combination with the weighting parameters $\alpha = \beta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$ (see Sect. 2.3). In Table 1, the optimal cost model for each manuscript is given.

Table 1. Optimal cost function parameter.

GW				PAR				AK				BOT			
τ_v	τ_e	α	β	τ_v	τ_e	α	β	τ_v	τ_e	α	β	τ_v	τ_e	α	β
8	4	0.1	0.5	8	1	0.5	0.1	16	1	0.1	0.3	8	4	0.1	0.3

Next, the proposed framework is tested using the same training and test sets as used in [3] and [6]. In Table 2, a summary of dataset characteristics of all four documents is given.

Table 2. Number of keywords, size of keyword spotting datasets (train and test), and the image resolution of the original documents in dpi.

Dataset	Keywords	Train	Test	dpi
GW	105	2447	1224	300
PAR	1217	11468	6869	200
AK	200	1849	3734	400
BOT	150	1684	3380	400

3.2 Cross-Evaluation

In Table 3, the results of the cross-evaluation are shown for all manuscripts (columns) using cross-evaluated parameters (rows). For instance, in the first row we show the KWS results on all four data sets of the system actually optimised on GW. In the main diagonal of Table 3 we thus provide the KWS results on a document obtained by the system optimised on the same document.

On GW we observe that the KWS system optimised on BOT achieves a similar result as the document specific system (actually this system performs even slightly better). Regarding the optimal parameters for the cost function in Table 1, this result makes sense as for both GW and BOT very similar parameters turn out to be optimal. Also the writing styles of both documents are quite similar (see, for instance, Fig. 1).

Likewise, we observe that on BOT the KWS system optimised on GW achieves quite similar results as the system optimised on BOT itself. In contrast with GW, however, we observe that on this dataset the parametrisation seems to have less influence on the KWS accuracy as all parametrisations lead to similar results. The same accounts for AK, where the optimal parameters for BOT turn out to achieve the best result on the test set.

On one document, viz. PAR, however, none of the systems optimised on another document can actually keep up with the system that has been optimised for this specific manuscript. That is we observe deteriorations of the KWS accuracy of about 6 to 12 basis points. The system optimised on PAR makes use of $\alpha = 0.5$, while all other systems turn out to be optimal with $\alpha = 0.1$. PAR has a more dense and straight (i.e. almost no slant) handwriting when compared to GW, AK, and BOT as shown in Fig. 1. As a result, variations in the x-direction become more relevant (thus the higher α value).

Table 3. MAP using optimised cost function parameters of one manuscript employed on all three remaining manuscripts. With ± we indicate the absolute percental gain or loss in the accuracy of the cross-evaluated manuscript when compared with the optimised parameter settings (shown in bold face).

Optimised on	GW		PAR		AK		BOT	
	MAP	±	MAP	±	MAP	±	MAP	±
GW	**69.28**	-	63.39	−5.84	79.54	−0.18	51.21	−0.48
PAR	64.84	−4.44	**69.23**	-	79.73	+0.01	51.12	−0.57
AK	61.45	−7.82	56.93	−12.30	**79.72**	-	50.81	−0.88
BOT	69.44	+0.17	62.40	−6.83	80.28	+0.56	**51.69**	-

Overall we conclude, that the weighting parameter α shows quite a strong correlation with the density of the handwriting. For a dense and straight handwriting the x-direction becomes more important, and thus higher parameter values for α should be chosen. In contrast to that, β has in most cases only a minor influence. Finally, it seems that the cost parameters τ_v and τ_e are depending on the size of the handwriting. That is, if the handwriting is characterised by flourish like in case of AK, for instance, node substitutions should be rather allowed by the cost model (by defining higher values for τ_v).

3.3 Ensemble Methods

In Table 4, we show the results of the proposed ensemble methods and the individually optimised systems for each document. In the first column, for instance, we show in the first row the KWS accuracy on GW of the system actually optimised on GW. The three ensemble methods combine the results of the three systems optimised on the remaining datasets.

In three out of four manuscripts, we observe that the ensemble methods can keep up or even improve the accuracy when compared with the individually optimised system. Especially, the ensemble methods max and mean achieve similar KWS accuracies without any a priori knowledge of the manuscript. Similar to the cross-evaluation experiment, we observe that ensemble methods can not keep up on PAR. In contrast to all other manuscripts PAR offers a different writing style, and the ensemble methods are not able to compensate these obvious differences.

Table 4. MAP of ensemble methods min, max, and mean. With ± we indicate the absolute percental gain or loss in the accuracy of the ensemble method when compared with the optimised parameter settings. Ensemble methods are ranked by (1), (2), (3).

	GW		PAR		AK		BOT		Average	
	MAP	±	MAP	±	MAP	±	MAP	±	MAP	±
Optimal	69.28		69.23		79.72		51.69		67.48	
min	65.63	−3.65 (3)	56.94	−12.29 (3)	80.28	+0.56 (2)	50.81	−0.88 (3)	63.42	−4.06 (3)
max	69.25	−0.02 (1)	63.39	−5.84 (1)	79.54	−0.18 (3)	51.25	−0.44 (2)	65.86	−1.62 (1)
mean	66.65	−2.62 (2)	62.28	−6.95 (2)	80.42	+0.70 (1)	51.49	−0.20 (1)	65.21	−2.27 (2)

We conclude that with ensemble methods (without document specific adaptations) similar KWS accuracy rates can be achieved as with individual document specific systems in most cases.

4 Conclusion and Outlook

The automatic recognition of handwritten historical documents is often negatively affected by noise (ink-bleed through, fading, degradation, etc.) as well as large handwriting variations in different documents. Therefore, *Keyword Spotting (KWS)* has been proposed as flexible and error-tolerant alternative to full transcriptions. Basically keyword spotting allows arbitrary retrievals in a document in order to make such a document accessible for browsing and searching.

In the last years, a number of graph-based keyword spotting approaches have been proposed. Yet, all of the proposed approaches are individually optimised and tested for each manuscript. Consequently, for a novel unseen document the system first needs to be optimised prior to the actual keyword spotting. In case of large collections or libraries this clearly reduces the overall applicability and practical relevance of the proposed graph-based keyword spotting frameworks.

In order to research this problem we conduct a cross-evaluation on four handwritten historical documents. That is, we evaluate KWS systems that have been optimised on other, unrelated documents. We observe a clear relationship between handwriting style and cost model for graph edit distance. Therefore, an unseen document could be directly accessed by a KWS system that has been optimised on a similar document without *a priori* parameter optimisation. Moreover, we show that ensemble methods allow to further increase the overall generalisability of the graph-based KWS. That is, the proposed ensemble methods, that need no document specific training, achieve similar accuracy rates as the optimised cost models.

In future work we aim at including further documents with different handwriting styles to our evaluation pipeline. Another research avenue to be pursued would be to research an automatic a priori *triage* in order to sort unknown manuscripts by means of their handwriting style. Finally, one could also extend the proposed ensemble methods by means of so-called *overproduce-and-select strategies*. That is, starting with the best individual system further cost model settings are added to an ensemble until a certain saturation is reached.

Acknowledgements. This work has been supported by the Swiss National Science Foundation project 200021_162852.

References

1. Ameri, M.R., Stauffer, M., Riesen, K., Bui, T.D., Fischer, A.: Graph-based keyword spotting in historical manuscripts using Hausdorff edit distance. Pattern Recognit. Lett. (2018, in press)
2. Bui, Q.A., Visani, M., Mullot, R.: Unsupervised word spotting using a graph representation based on invariants. In: International Conference on Document Analysis and Recognition, pp. 616–620. IEEE (2015)
3. Fischer, A., Keller, A., Frinken, V., Bunke, H.: Lexicon-free handwritten word spotting using character HMMs. Pattern Recognit. Lett. **33**(7), 934–942 (2012)
4. Fischer, A., Suen, C.Y., Frinken, V., Riesen, K., Bunke, H.: Approximation of graph edit distance based on Hausdorff matching. Pattern Recognit. **48**(2), 331–343 (2015)
5. Manmatha, R., Han, C., Riseman, E.: Word spotting: a new approach to indexing handwriting. In: Computer Society Conference on Computer Vision and Pattern Recognition, pp. 631–637. IEEE (1996)
6. Pratikakis, I., Zagoris, K., Gatos, B., Puigcerver, J., Toselli, A.H., Vidal, E.: ICFHR2016 handwritten keyword spotting competition (H-KWS 2016). In: International Conference on Frontiers in Handwriting Recognition, pp. 613–618. IEEE (2016)
7. Riba, P., Fischer, A., Lladós, J., Fornés, A.: Learning graph distances with message passing neural networks. In: International Conference on Pattern Recognition. IEEE (2018)
8. Riba, P., Lladós, J., Fornés, A.: Handwritten word spotting by inexact matching of grapheme graphs. In: International Conference on Document Analysis and Recognition, pp. 781–785. IEEE (2015)

9. Riesen, K., Bunke, H.: Approximate graph edit distance computation by means of bipartite graph matching. Image Vis. Comput. **27**(7), 950–959 (2009)
10. Stauffer, M., Fischer, A., Riesen, K.: A novel graph database for handwritten word images. In: Robles-Kelly, A., Loog, M., Biggio, B., Escolano, F., Wilson, R. (eds.) S+SSPR 2016. LNCS, vol. 10029, pp. 553–563. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49055-7_49
11. Stauffer, M., Fischer, A., Riesen, K.: Ensembles for graph-based keyword spotting in historical handwritten documents. In: International Conference on Document Analysis and Recognition, pp. 714–720. IEEE (2017)
12. Stauffer, M., Fischer, A., Riesen, K.: Keyword spotting in historical handwritten documents based on graph matching. Pattern Recognit. **81**, 240–253 (2018)
13. Wang, P., Eglin, V., Garcia, C., Largeron, C., Lladós, J., Fornés, A.: A novel learning-free word spotting approach based on graph representation. In: International Workshop on Document Analysis Systems, pp. 207–211. IEEE (2014)