

Annotation-free Learning of Deep Representations for Word Spotting using Synthetic Data and Self Labeling

Fabian Wolf^[0000-0001-8842-3718] and Gernot A. Fink^[0000-0002-7446-7813]

TU Dortmund University,
Department of Computer Science,
44227 Dortmund, Germany
{`firstname.lastname`}@cs.tu-dortmund.de

Abstract. Word spotting is a popular tool for supporting the first exploration of historic, handwritten document collections. Today, the best performing methods rely on machine learning techniques, which require a high amount of annotated training material. As training data is usually not available in the application scenario, *annotation-free* methods aim at solving the retrieval task without representative training samples. In this work, we present an *annotation-free* method that still employs machine learning techniques and therefore outperforms other *annotation-free* approaches. The weakly supervised training scheme relies on a lexicon, that does not need to precisely fit the dataset. In combination with a confidence based selection of pseudo-labeled training samples, we achieve state-of-the-art *query-by-example* performances. Furthermore, our method allows to perform *query-by-string*, which is usually not the case for other *annotation-free* methods.

Keywords: Word Spotting · Annotation-free · Weakly supervised.

1 Introduction

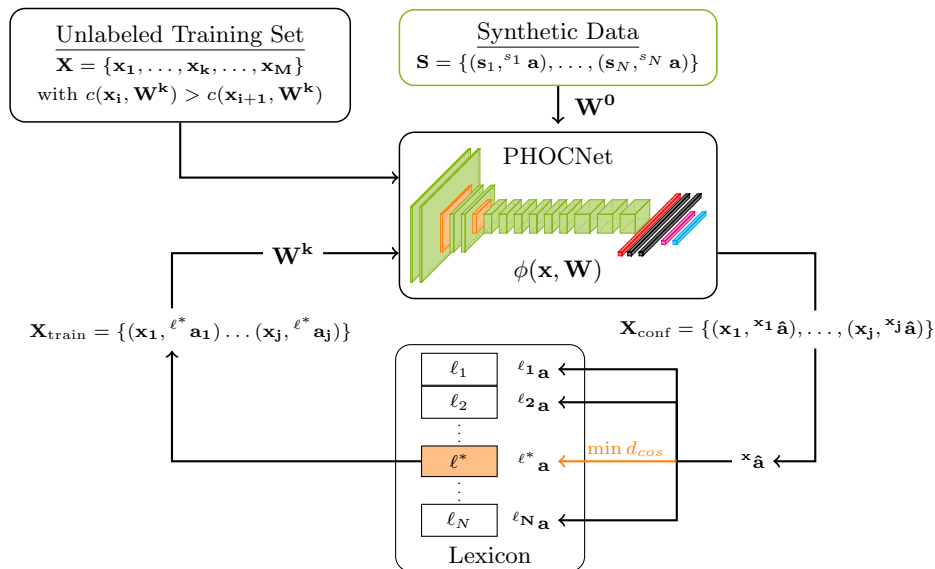
The digitization of documents sparked the creation of huge digital document collections that are a massive source of knowledge. Especially, historic and handwritten documents are of high interest for historians. Nonetheless, information retrieval from huge document collections is still cumbersome. Basic functionalities such as an automatic search for word occurrences are extremely challenging, due to the high visual variability of handwriting and degradation effects. Traditional approaches like *optical character recognition* often struggle when it comes to historic collections. In these cases, word spotting methods that do not aim at transcribing the entire document offer a viable alternative [8]. Word spotting describes the retrieval task of finding the most probable occurrences of a word of interest in a document collection. As the system provides a ranked list of alternatives, it is up to the expert and his domain knowledge to decide which entities are finally relevant.

Considering document analysis research, machine learning strongly influenced word spotting methods and a multitude of systems emerged [8]. Common taxonomies distinguish methods based on the type of query representation, a previously or simultaneously performed segmentation step and the necessity of a training procedure. Most systems represent the query either by an exemplar image (*query-by-example*) [19–21] or a string representation (*query-by-string*) [13, 27, 32]. In order to localize a query, the documents need to be segmented into word images. *Segmentation-based* methods [13, 20, 27] assume that this segmentation step is performed independently beforehand. In contrast, *segmentation-free* methods such as [22, 33] aim at solving the retrieval and segmentation problem jointly. Another distinction commonly made concerns the use of machine learning methods. So called *learning-free* techniques rely on expert designed feature representations [20, 30] and they are usually directly applicable as they do not rely on a learning phase. Motivated by the success in other computer vision tasks, machine *learning-based* techniques and especially convolutional neural networks dominate the field of word spotting today [27, 28, 33].

The distinction between *learning-free* and *learning-based* word spotting methods suggests that applying machine learning methods is a disadvantage in itself. This is only true for supervised learning approaches that require huge amounts of annotated training material in order to be successful. In cases where learning can be applied without such a requirement, we can not see any disadvantage of leveraging the power of machine learning for estimating models of handwriting for word-spotting purposes. We therefore suggest to distinguish methods based on the requirement of training data. Methods that do not rely on any manually labeled samples will be termed *annotation-free* as opposed to *annotation-based* techniques relying on supervised learning, as most current word spotting approaches based on deep learning do [27, 28]. Today, almost all *annotation-free* methods are also *learning-free* as it is not straight forward to devise a successful learning method that can be applied if manual annotations are not available. These *learning-free* methods provide a feature embedding that encodes the visual appearance of a word [20, 30]. As no model for the appearance of handwriting is learned, *query-by-string* is usually out of scope for these approaches.

In this work, we propose an *annotation-free* method for *segmentation-based* word spotting that overcomes this drawback by performing learning without requiring any manually labeled data. The proposed method uses a synthetic dataset to train an initial model. Due to the supervised training on the synthetic dataset, the model is capable to perform *query-by-string* word spotting. This initial model is then transferred to the target domain iteratively in a semi-supervised manner. Our method exploits the use of a lexicon which is used to perform word recognition to generate pseudo-labels for the target domain. The selection of pseudo-labels used to train the network is based on a confidence measure. We show that a confidence based selection is superior to randomly selecting training samples and already a rough estimate of the lexicon is sufficient to outperform other *annotation-free* methods. The proposed training scheme is summarized in Fig. 1.

Fig. 1. Semi-supervised training scheme: First an initial model is trained on synthetic data. The model is then iteratively transferred to the target domain by training on confidently estimated samples which are pseudo-labeled with lexicon based recognition.



2 Related Work

In order to solve the retrieval task of word spotting, a system evaluates the similarity between document image regions and a query. Similar to popular recognition models, many methods exploit the sequential structure of handwriting. In an early work on *segmentation-based* word spotting, Rath and Manmatha proposed to use dynamic time warping to quantify the similarity of two word images based on the optimal alignment of their sequences [19]. Other sequential models such as *hidden Markov models* (HMM) [21, 31] and *recurrent neural networks* [17] were also successfully used for word spotting and they are still popular today [28].

Traditional feature extraction methods also saw high popularity, due to their success in other computer vision tasks. In this case, the general approach is to embed the visual appearance of a word image in a feature vector. This *holistic* representation can then be easily compared to other document regions or image queries with a simple distance measure. Traditional descriptors based on gradient orientations such as HOG, LBP and SIFT have been shown to be suitable to capture the characteristics of handwriting [2, 10, 20]. Usually the descriptors of an image patch are accumulated in a histogram following the idea of a *Bag of Visual Words (BoVW)*. Since such a histogram vector neglects any spatial relations between descriptors, it is necessary to combine the approach with an

additional model. For example, [1] and [24] use a pyramidal scheme to add spatial information, while [21] encodes a sequence of BoVW vectors with an HMM. As these feature-based approaches only embed the visual appearance, they often struggle when spotting is performed across multiple writing styles. Queries need to be given in the image domain, which only allows *query-by-example* word spotting.

These limitations motivated the use of *learning-based* approaches. In an influential work [3], Almazan et al. proposed the use of attribute representations. The method aims at learning the mapping between word images and a *Pyramidal Histogram of Characters*, which is a binary vector encoding the spatial occurrences of characters. As the derivation of a PHOC vector from a string is trivial, word images and strings can be mapped in a common embedding space, allowing *query-by-string*. In [3], the visual appearance of a word image is first encoded in a Fisher Vector, followed by a set of Support Vector Machines predicting the presence or absence of each attribute.

The application of neural networks and deep learning also resulted in a strong performance gain in the field of word spotting. In [26], a convolutional neural network is employed to learn attribute representations similar to [3]. Other methods such as [12] or [32] also employed neural networks to learn different feature embeddings. Essentially, the proposed methods based on neural networks significantly outperformed all previous approaches for *query-by-example* and *query-by-string*. While the discussed networks are usually trained on segmented word images, it has been shown that the approaches can be effectively adapted to the *segmentation-free* scenario by using word hypotheses [22] or region proposal networks [33].

Although *learning-based* methods showed exceptional performances on almost all benchmarks, they rely on a tremendous amount of training data. Considering the application of a word spotting tool, which is the exploration of a so far unknown document collection, the assumption that annotated documents are available rarely holds. This problem is far from being exclusive to word spotting and it has been of interest to the computer vision and machine learning communities in general. *Semi-supervised learning* describes the concept of using unlabeled data in combination with only a limited amount of annotated training samples [5]. In [6], this approach was successfully applied for a document analysis tasks. The authors used a word spotting system on a unlabeled dataset to generate additional labeled samples for a handwriting recognition system. A special type of *semi-supervised* methods employ so called self-labeling techniques [29], which have been also studied for neural networks [14]. In general, an initial model is trained on an annotated dataset and later used to generate labels for unlabeled data of the target domain. These pseudo-labeled data samples are integrated in the training scheme to further adapt the model.

Transfer-learning describes another approach to reduce the need of training data. It has been shown that data from another domain can be used to efficiently pre-train a model. In [13], a synthetic dataset for training word spotting models is proposed. Annotated training samples are rendered from computer fonts that

resemble handwriting. The resulting dataset is used for pre-training a network that is then fine tuned on samples from the target domain. As shown in [9], training a model exclusively on synthetic data does not allow for state of the art performances. Anyways, the amount of training data necessary to achieve competitive results can be reduced significantly.

The lack of training data is a crucial problem for word spotting on historic datasets. While being clearly outperformed by fully-supervised methods and missing the possibility to perform *query-by-string* word spotting, *annotation-free*, feature-based methods are still receiving attention from the research community [20,30].

3 Method

Our method evolves around a basic word spotting system based on an attribute CNN. We use the TPP-PHOCNet architecture proposed in [27] to estimate the attribute representation of an input word image. A 4-level PHOC representation of partitions 1,2,4 and 8 serves as a word string embedding. In all experiments, the assumed alphabet is the Latin alphabet plus digits, which results in an attribute vector $\mathbf{a} \in (0, 1)^D$ with $D = 540$. Given a trained network the system allows to perform word spotting and lexicon-based word recognition, as described in Sec. 3.1.

The proposed training scheme presented in Sec. 3.2 does not require any manually annotated training material. Starting with an initial model, the system exploits the use of automatically generated pseudo-labels for the target domain. In order to enhance the accuracy of the generated labels, only a subset of the predicted pseudo-labels is used during the next training cycle. The selection of samples is based on an estimate of how confident the network is in its predictions. In this work, we compare the three confidence measures described in Sec. 3.3.

3.1 Word Spotting and Recognition

Given a trained PHOCNet with weights \mathbf{W} , the network constitutes a function ϕ that estimates the desired attribute representation $\hat{\mathbf{a}} = \phi(\mathbf{x}, \mathbf{W})$ for an input word image \mathbf{x} . In the segmentation-based scenario, word spotting is then performed by ranking all word images of the document according to their similarity to a query. In this work, similarity is measured by the cosine dissimilarity d_{cos} between the estimated attribute vector and the query vector. Depending on the query paradigm, the query vector ${}^q\mathbf{a}$ is either directly derived from a query string or estimated from a query word image \mathbf{q} with ${}^q\hat{\mathbf{a}} = \phi(\mathbf{q}, \mathbf{W})$.

Word recognition is performed in a similar manner. Let \mathbb{L} be a lexicon of size N with a set of corresponding attribute representations ${}^{\ell_i}\mathbf{a}$, $i \in 1, \dots, N$. Based on the estimated attribute vector ${}^x\hat{\mathbf{a}}$ for the input image \mathbf{x} , word recognition reduces to a nearest neighbour search over the lexicon. Therefore, the recognition result ℓ^* is given by:

$$\ell^* = \underset{\ell \in \mathbb{L}}{\operatorname{argmin}} d_{cos}({}^{\ell}\mathbf{a}, {}^x\hat{\mathbf{a}}). \quad (1)$$

Algorithm 1: Semi-supervised training procedure

Input : Synthetic data $\mathbf{S} = \{(\mathbf{s}_1, \mathbf{a}^{s_1}), \dots, (\mathbf{s}_N, \mathbf{a}^{s_N})\}$; unlabeled training images $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_M\}$; number of training cycles K ; PHOCNet $\phi(\cdot, \mathbf{W})$; confidence measure $c(\cdot, \mathbf{W})$;

- 1 Train initial model $\phi(\cdot, \mathbf{W}^0)$ on \mathbf{S} ;
- 2 **for** $k \leftarrow 0$ **to** K **do**
- 3 Estimate attribute representation ${}^x\hat{\mathbf{a}} = \phi(\mathbf{x}, \mathbf{W}^k)$ for each element \mathbf{x} in \mathbf{X} ;
- 4 Sort \mathbf{X} w.r.t. confidence: $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_j, \dots, \mathbf{x}_M\}$ with $c(\mathbf{x}_i, \mathbf{W}^k) > c(\mathbf{x}_{i+1}, \mathbf{W}^k)$;
- 5 Select j most confident samples $\mathbf{X}_{conf} = \{\mathbf{x}_1, \dots, \mathbf{x}_j\}$;
- 6 Generate pseudo labels with word recognition $\mathbf{X}_{train} = \{(\mathbf{x}_1, \mathbf{a}^{\ell_1^*}), \dots, (\mathbf{x}_j, \mathbf{a}^{\ell_j^*})\}$;
- 7 $\mathbf{W}^{k+1} \leftarrow \text{train } \phi(\cdot, \mathbf{W}^k)$ on \mathbf{X}_{train}

3.2 Training Scheme

The proposed training scheme is summarized in algorithm 1. Initially, we train the model $\phi(\cdot, \mathbf{W})$ on a purely synthetically generated dataset. Its images \mathbf{s}_i are generated based on computer fonts that resemble handwriting. The corresponding attribute representations \mathbf{a}^{s_i} are known and their creation does not cause any manual annotation effort.

In order to further improve the initial model $\phi(\cdot, \mathbf{W}^0)$, we exploit the unlabeled target dataset. First, an estimate of the attribute representation ${}^x\hat{\mathbf{a}}$ is computed for each word image \mathbf{x} in the target dataset \mathbf{X} . As shown in [9], a model only trained on synthetic data does not yield a very high performance and it is likely that the estimated attribute vectors are highly inaccurate. In order to still derive a reliable pseudo-label, unreliable samples are removed and a lexicon serves as an additional source of domain information.

Inaccurate estimates of attribute vectors are identified by the use of a confidence measure. Essentially, a confidence measure constitutes a function $c(\cdot, \mathbf{W})$ that quantifies the quality of the network outputs based on its current weights \mathbf{W} . In this work, we investigate three different approaches to measure the network’s confidence, see Sec. 3.3.

In each cycle a fixed percentage of confidently estimated samples is selected. For each sample in this confident part of the unlabeled dataset $\mathbf{X}_{conf} = \{\mathbf{x}_1, \dots, \mathbf{x}_j\}$ a pseudo-label is generated. The label l^* is derived by performing word recognition with the lexicon \mathbb{L} , as described in Sec. 3.1. Therefore, the attribute vector representation used as a training target is derived from the lexicon entry with minimal cosine dissimilarity to the estimated attribute representation ${}^x\hat{\mathbf{a}}$. The resulting dataset with pseudo-labels is then used for training in order to further adapt the model.

The process of estimating attribute representations based on the current state of the model, selecting confident samples and generating pseudo-labels is performed repeatedly for K cycles. As the training set of pseudo-labeled data

\mathbf{X}_{train} is comparably small and the model therefore prone to overfitting the same regularization techniques as proposed in [27] are used. The set of training samples is augmented using random affine transformations. Based on the predicted labels, word classes are balanced in the resulting augmented training set. As an additional regularization measure, the PHOCNet architecture employs dropout in its fully connected layers.

3.3 Confidence Measures

As shown in [34], confidence measures allow to quantify the quality of an attribute vector prediction. Furthermore, recognition accuracies are higher on more confident parts of a dataset, making it a suitable tool for pseudo-label selection. Following the approach of [23], we model each Attribute A_i as a binary random variable following a Bernoulli distribution. The output of the attribute CNN is given by ${}^x\hat{\mathbf{a}} = \phi(\mathbf{x}, \mathbf{W})$. Each element of the output vector is considered an estimate ${}^x\hat{a}_i \approx p(A_i = 1|\mathbf{x})$ for the probability of the i -th attribute being present in the word image \mathbf{x} .

Sigmoid Activation Based on the work in [34], we derive a confidence measure directly from the network outputs. Each sigmoid activation provides a pseudo probability for the estimate \hat{a}_i . To estimate the confidence of an entire attribute vector we follow the approach of [34] and we sum over the estimates of all active attributes. Neglecting inactive attributes, i.e., attributes with a pseudo probability of $\hat{a}_i < 0.5$, resulted in a slightly better performance in our experiments. We believe that this is due to the estimated attribute vector being almost binary with only a few attributes close to one. It seems that the confidence estimation for the high number of absent attributes disturbs the overall assessment of the attribute vector. The resulting confidence measure $c(\mathbf{x}, \mathbf{W})$ is then given by

$$c(\mathbf{x}, \mathbf{W}) = \sum_{{}^x\hat{a}_i > 0.5} \phi(\mathbf{x}, \mathbf{W})_i \approx \sum_{{}^x\hat{a}_i > 0.5} p(A_i = 1|\mathbf{x}). \quad (2)$$

Test Dropout Another approach to estimate uncertainty is to use dropout as an approximation [7]. A confidence measure can be derived by applying dropout layers at test time. By performing multiple forward passes a variance can be observed for each attribute estimate \hat{a}_i . In this case the assumption is that for a confident prediction the estimate remains constant although neurons are dropped in the dropout layers. The approach is directly applicable to the PHOCNet as shown in [34]. All fully connected layers except the last one are applying dropout with a probability of 0.5. We calculate the mean over all attribute variances over 100 forward passes. A high confidence corresponds to a low mean attribute variance.

Entropy A well known concept from information theory is to use entropy to measure the amount of information received by observing a random variable

[4]. The observation of a random variable with minimal entropy does not hold any information. Therefore, there is no uncertainty about the realization of the random variable. In this case, a low entropy corresponds to a high confidence in the network’s predictions. Following the interpretation of an attribute as a Bernoulli distributed random variable A_i , its entropy is given by

$$H(A_i) = -\hat{a}_i \log \hat{a}_i - (1 - \hat{a}_i) \log(1 - \hat{a}_i). \quad (3)$$

To model the confidence of an entire attribute vector, we compute the negative joint entropy over all attributes. As in [23], we assume conditional independence among attributes. The joint entropy over all attributes is then computed by the sum over the entropies of the individual random variables.

$$\begin{aligned} c(\mathbf{x}, \mathbf{W}) &= -H(A_1, \dots, A_D) = -\sum_{i=1}^D H(A_i) \\ &= \sum_{i=1}^D \hat{a}_i \log \hat{a}_i + (1 - \hat{a}_i) \log(1 - \hat{a}_i). \end{aligned} \quad (4)$$

4 Experiments

We evaluate our method on four benchmark datasets for *segmentation-based* word spotting. In those cases where an annotated training set is available, we do not make use of any labels. For details on the datasets and the specific evaluation protocols see Sec. 4.1. We use *mean average precision* (mAP) in all our experiments to measure performance and to allow for a direct comparison to other methods [8]. As the provision of an exact lexicon can be quite a limitation in an application scenario, Sec. 4.3 presents experiments on different choices of lexicons. Sec. 4.4 provides an evaluation of different confidence measures and investigates the question whether a confidence based selection of samples is superior to random sampling. In Sec. 4.5, we compare our method to the state-of-the-art and especially to *annotation-free* methods.

4.1 Datasets

George Washington The George Washington (GW) dataset has been one of the first datasets used to evaluate *segmentation-based* word spotting [19]. The documents were published by the Library of Congress, Washington DC, USA and they contain letters written by George Washington and his secretaries. In general, the writing style of the historic dataset is rather homogeneous. The benchmark contains 4860 segmented and annotated word images. As no distinctive separation in training and test partition exist, we follow the four-fold cross validation protocol presented in [3]. Although we train our network on the training splits, we do not make use of the annotations. Images from the test split are considered to represent unknown data and are therefore only used for evaluation.

IAM The IAM database was created to train and to evaluate handwriting recognition models [15]. 657 different writers contributed to the creation of the benchmark, leading to a huge variety of writing styles. In total over 115000 annotated word images are split into a training, validation and test partition. No writer contributed to more than one partition. Due to its size and the strong variations in writing styles the IAM database became another widespread benchmark for word spotting [8]. The common approach for word spotting is to use each word image (*query-by-example*) or each unique transcription (*query-by-string*) of the test set as a query once. Stop words are not used as queries but still kept in the test set as distractors.

Bentham The Bentham datasets originated from the project *Transcribe Bentham* and were used for the keyword spotting competitions at the *International Conference on Frontiers in Handwriting Recognition 2014* (BT14) [16] and at the *International Conference on Document Analysis and Recognition 2015* (BT15) [18]. The historic datasets contain documents written by the English philosopher Jeremy Bentham and show some considerable variations in writing styles. Both competitions define a *segmentation-based, query-by-example* benchmark. The BT14 set consists of 10370 segmented word images and a set of 320 designated queries. For BT15 the dataset was extended to 13657 word images and a significantly larger number of queries of 1421.

IIIT-HWS We use a synthetically generated dataset to train an initial model without any manual annotation effort. The IIIT-HWS dataset, proposed in [13], was created from computer fonts that resemble handwriting. Based on a dictionary containing 90000 words, a total number of 1 million word images were created and successfully used to train a word spotting model. We use the published dataset to train our models and did not make any changes to the generation process.

4.2 Training Details

As the proposed method is based on the TPP-PHOCNet architecture, we mainly stick to the hyperparameters that have been proven successful in [27]. We train the network in an end to end fashion with Binary Cross Entropy and the ADAM optimizer. All input word images are inverted, such that the actual writing, presumably dark pixels, is represented by a value of one. In all experiments, we use a batch size of 10, weight decay of $5 \cdot 10^{-5}$ and we employ a momentum with mean 0.9 and variance 0.999.

Our model is initially trained on the IIIT-HWS dataset for 70000 iterations with a learning rate of 10^{-4} , followed by another 10000 training iterations with a learning rate of 10^{-5} . We follow the approach of [9] and randomly resize the synthetic word images during training to cope with differently sized images in the benchmark datasets. Each synthetic word image is scaled by a random factor within the interval $[1, 2)$.

Table 1. Evaluation of different lexicons. Pseudo-labels are selected randomly in all cases. Results reported as mAP [%].

Lexicon	GW		IAM		BT14		BT15	
	QbE	QbS	QbE	QbS	QbE	QbS	QbE	QbS
None (*)	46.6	57.9	16.0	39.5	18.1	-	16.4	-
Language Based	73.1	64.0	56.9	77.1	79.2	-	65.2	-
Closed	87.8	87.8	63.7	83.6	-	-	-	-
Bentham	-	-	-	-	84.3	-	69.1	-

(*) initial model, no weakly supervised training.

The following training phase, which is only weakly supervised by a lexicon, is performed in multiple cycles. After each cycle, old pseudo-labels are neglected and a new set is generated with word recognition and the selection scheme. On all datasets except IAM we create a total number of 10000 samples using the augmentation method presented in [26]. Word classes are balanced based on the pseudo-labels. Due to the bigger size of the IAM database, we augment the pseudo-labeled samples to 30000 images. For each cycle, the network is trained for one epoch with respect to the augmented training set and a learning rate of 10^{-5} . In our experiments, we train the network for $K = 20$ cycles. After selecting 10% of the pseudo-labels as training samples during the first 10 cycles we increase the percentage to 60%.

4.3 Lexicon

In a first set of experiments, we investigate how crucial the prior knowledge on the lexicon is. All experiments do not make use of a confidence measure but perform the selection of pseudo-labeled samples randomly. We experiment with three different types of lexicons and compare the results against the performance of the network after training only on synthetic data. First we assume that only the language of the document collection is known. To derive a lexicon for all our datasets, we use the 10000 most common English words. Note that this results in 13.5% out-of-vocabulary words on GW, and 10.4% on the IAM database. Due to the lack of transcriptions, we cannot report out of vocabulary percentages for the Bentham datasets. As all samples in the GW and IAM datasets are labeled, we can create a closed lexicon containing all training and test transcriptions of the respective datasets. Even though, this is the most precise lexicon resulting in no out-of-vocabulary words, we argue that in an application scenario an exact lexicon is usually not available. In case of the Bentham datasets, we investigate another lexicon that is based on the manual line-level annotations published in [18]. This resembles the case that some related texts, potentially written by the same author, are available and provide a more precise lexicon.

Table 1 presents the resulting spotting performances with respect to the different lexicons. In general, performances increase substantially by training

Table 2. Evaluation of confidence measures. All experiments use a language based lexicon. Results reported as mAP [%]. Best *annotation-free* results are marked in bold.

Confidence	GW		IAM		BT14		BT15	
	QbE	QbS	QbE	QbS	QbE	QbS	QbE	QbS
Random	73.1	64.0	56.9	77.1	79.2	-	65.2	-
Entropy	79.5	82.1	62.6	81.3	84.2	-	75.2	-
Sigmoid	83.2	82.3	62.6	81.0	87.2	-	76.3	-
Test Dropout	50.4	39.7	19.5	34.3	23.4	-	18.6	-
$d_{\cos}({}^x\hat{\mathbf{a}}, {}^t\mathbf{a})$	93.8	94.3	75.0	87.7	-	-	-	-

on the handwritten samples from the target domain under weak supervision. Already the approximate lexicon based on the modern English language results in high performance gains also for the historic benchmarks. For the closed as well as the related Bentham lexicon, it can be seen that performances increase with a more precise lexicon. Nonetheless, we would argue that in the considered scenario only a language based lexicon, which does not require any additional information on the texts besides their language, is a reasonable option.

4.4 Confidence Measures

As discussed in [Sec. 3.3](#), a confidence measure can be used to identify parts of a dataset that have higher recognition accuracies. In our experiments, we use the three approaches described in [Sec. 3.3](#) to quantify confidence. We only select the most confident pseudo-labeled samples to continue training. Furthermore, we conducted another experiment that uses the cosine dissimilarity between the estimated attribute representation ${}^x\hat{\mathbf{a}}$ and the actual transcription ${}^t\mathbf{a}$ as a confidence measure. This is motivated by the idea that a confidence measure essentially quantifies the quality of the attribute estimation, which corresponds to the similarity between estimation and annotation. Although in practice the cosine dissimilarity cannot be computed without a given annotation, it gives us an upper bound on how well the method would perform with a perfect confidence estimation.

[Table 2](#) presents the results of the experiments, which are conducted with the different confidence measures. For entropy and sigmoid activations, we observe a performance gain on all benchmarks compared to a random sample selection. Despite the clear probabilistic interpretation, using entropy performs only on par with sigmoid activations and it is slightly outperformed on the presumably simpler datasets of GW and BT14. The use of test dropout does not yield any satisfactory results and even performs worse than a random approach. We observed that test dropout only gives high confidences for rather short words, which makes the selected pseudo-labeled samples not very suitable as training samples. A longer word potentially provides a bigger set of correct annotation on the attribute level, even in cases where the pseudo-label is wrong.

Table 3. Comparison on GW and IAM. Results reported as mAP [%]. Best *annotation-free* results are marked in bold, best overall in italic.

Method	Annotations [n]	GW		IAM	
		QbE	QbS	QbE	QbS
Languaged Based & Sigmoid	0	83.2	82.3	62.6	81.0
Almazan et al. [2]	0	49.4	-	-	-
Sfikas et al. [25]	0	58.3	-	13.2	-
DTW [3]	0	60.6	-	12.3	-
FV [3]	0	62.7	-	15.6	-
Retsinas et al. [20]	0	77.1	-	28.1	-
Gurjar et al. [9]	0	39.8	48.9	26.2	36.5
Gurjar et al. [9]	1000	95.7	96.5	55.3	74.0
AttributeSVM [3]	complete	93.0	91.2	55.7	73.7
TPP-PHOCNet [27]	complete	97.9	96.7	84.8	92.9
STPP-PHOCNet [23]	complete	97.7	96.8	89.2	<i>95.4</i>
Deep Embed [11]	complete	<i>98.0</i>	<i>98.8</i>	<i>90.3</i>	94.0
Triplet-CNN [32]	complete	98.0	93.6	81.5	89.4

Considering the use of cosine dissimilarity, it can be seen that a more accurate confidence estimation can still improve performance. The proposed method in combination with cosine dissimilarity outperforms all other confidence measures, suggesting that the proposed confidence measures are providing suboptimal estimates only.

4.5 Comparison

In order to allow for a fair comparison to the state-of-the-art we only consider the performance of our method with respect to sigmoid activation as a confidence measure and a language based lexicon. Compared to other *annotation-free* methods, the only additional prior knowledge which we exploit, is the language of the considered documents. Table 3 reports the performance of the proposed method and other *annotation-free* and *annotation-based* approaches on the GW and IAM dataset. The best results so far that do not require training material are reported in [20]. Our method achieves higher mean average precisions on both datasets. Note that the difference is substantially higher in case of the IAM database. The work in [20] is heavily based on a specific feature design to incorporate visual appearance, which is quite suitable for the homogeneous appearance of the GW dataset. Nonetheless, our method outperforms all other *annotation-free* methods, while the difference is more substantial on datasets as the IAM database where writing styles and visual appearance vary strongly.

In [9], experiments were presented that show how performance increases, when a limited number of annotated samples is used to fine tune a network

Table 4. Comparison for the *annotation-free, query-by-example* benchmark on the Bentham datasets. Results reported as mAP [%]. Best results are marked in bold.

Method	BT14	BT15
	QbE	QbE
Languaged Based & Sigmoid	87.2	76.3
Aldavert et al. [1]	46.5	-
Almazan et al. [3]	51.3	-
Kovalchuk et al. [10]	52.4	-
CVC [18]	-	30.0
PRG [18]	-	42.4
Sfikas et al. [25]	53.6	41.5
Zagoris et al. [35]	60.0	50.1
Retsinas et al. [20]	71.1	58.4

similar to ours. While a number of 1000 annotated samples are sufficient to outperform our semi-supervised approach on GW, we still achieve better performances on IAM. This suggests that our model is able to learn characteristics across different writing styles without relying on any annotations. Due to the lack of annotated training data from the target domain, our method performs worse compared to fully supervised approaches.

The experiments on both Bentham datasets reported in Table 4 support our observations. As the benchmarks are considered *annotation-free*, no word image labels are provided. Therefore, we cannot report any quantitative evaluation of *query-by-string* word spotting. While outperforming all other methods in the *query-by-example* case, our method additionally offers the possibility to perform *query-by-string*, which is not the case for all other *annotation-free* approaches.

5 Conclusions

In this work, we show that an *annotation-free* method for *segmentation-based* word spotting, which does not use any manually annotated training material, can still successfully employ machine learning techniques. Compared to other methods that do not include a learning phase, this leads to significant improvements in performance. The proposed method relies on a lexicon that provides additional domain information. Our experiments show that already a language based lexicon, which does not necessarily precisely correspond to the considered documents, is sufficient to achieve state-of-the-art performances. We successfully make use of a confidence measure to select pseudo-labeled samples during training to boost overall performance. Additionally, our method provides the capability to perform *query-by-string* word spotting, which is usually not the case for other *annotation-free* approaches. Therefore, our method is highly suitable for the exploration of heterogeneous datasets where no training material is available.

References

1. Aldavert, D., Rusiñol, M., Toledo, R., Lladós, J.: A study of bag-of-visual-words representations for handwritten keyword spotting. *Int. Journal on Document Analysis and Recognition* **18**(3), 223–234 (2015)
2. Almazán, J., Gordo, A., Fornés, A., Valveny, E.: Efficient exemplar word spotting. In: *British Machine Vision Conf. Surrey, UK* (2012)
3. Almazán, J., Gordo, A., Fornés, A., Valveny, E.: Word spotting and recognition with embedded attributes. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **36**(12), 2552–2566 (2014)
4. Bishop, C.M.: *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)
5. van Engelen, J.E., Hoos, H.H.: A survey on semi-supervised learning. *Machine Learning* (Nov 2019)
6. Frinken, V., Baumgartner, M., Fischer, A., Bunke, H.: Semi-supervised learning for cursive handwriting recognition using keyword spotting. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 49–54. Bari, Italy (2012)
7. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. New York City, NY, USA (2016)
8. Giotis, A.P., Sfikas, G., Gatos, B., Nikou, C.: A survey of document image word spotting techniques. *Pattern Recognition* **68**, 310–332 (2017)
9. Gurjar, N., Sudholt, S., Fink, G.A.: Learning deep representations for word spotting under weak supervision. In: *Proc. Int. Workshop on Document Analysis Systems*. pp. 7–12. Vienna, Austria (2018)
10. Kovalchuk, A., Wolf, L., Dershowitz, N.: A simple and fast word spotting method. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 3–8. Crete, Greece (2014)
11. Krishnan, P., Dutta, K., Jawahar, C.V.: Word spotting and recognition using deep embedding. In: *Proc. Int. Workshop on Document Analysis Systems*. pp. 1–6. Vienna, Austria (2018)
12. Krishnan, P., Dutta, K., Jawahar, C.: Deep feature embedding for accurate recognition and retrieval of handwritten text. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 289 – 294. Shenzhen, China (2016)
13. Krishnan, P., Jawahar, C.V.: Hwnet v2: an efficient word image representation for handwritten documents. *Int. Journal on Document Analysis and Recognition* **22**(4), 387–405 (2019)
14. Lee, D.H.: Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. *ICML 2013 Workshop: Challenges in Representation Learning (WREPL)* (2013)
15. Marti, U., Bunke, H.: The IAM-database: an english sentence database for offline handwriting recognition. *Int. Journal on Document Analysis and Recognition* **5**(1), 39–46 (2002)
16. Pratikakis, I., Zagoris, K., Gatos, B., Louloudis, G., Stamatopoulos, N.: ICFHR 2014 competition on handwritten keyword spotting (H-KWS 2014). In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 814–819. Crete, Greece (2014)
17. Puigcerver, J.: A probabilistic formulation of keyword spotting. Dissertation, Universitat Politècnica de València, València, Spain (2018)
18. Puigcerver, J., Toselli, A., Vidal, E.: Icdar2015 competition on keyword spotting for handwritten documents. In: *Proc. Int. Conf. on Document Analysis and Recognition*. pp. 1176–1180. Nancy, France (2015)

19. Rath, T.M., Manmatha, R.: Word spotting for historical documents. *Int. Journal on Document Analysis and Recognition* **9**(2-4), 139–152 (2007)
20. Retsinas, G., Louloudis, G., Stamatopoulos, N., Gatos, B.: Efficient learning-free keyword spotting. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **41**(7), 1587–1600 (2019)
21. Rothacker, L.: Segmentation-free word spotting with bag-of-features hidden Markov models. Dissertation, TU Dortmund University, Dortmund, Germany (2019)
22. Rothacker, L., Sudholt, S., Rusakov, E., Kasperidus, M., Fink, G.A.: Word hypotheses for segmentation-free word spotting in historic document images. In: *Proc. Int. Conf. on Document Analysis and Recognition*. Kyoto, Japan (2017)
23. Rusakov, E., Rothacker, L., Mo, H., Fink, G.A.: A probabilistic retrieval model for word spotting based on direct attribute prediction. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 38–43. Niagara Falls, NY, USA (2018)
24. Rusiñol, M., Aldavert, D., Toledo, R., Lladós, J.: Efficient segmentation-free keyword spotting in historical document collections. *Pattern Recognition* **48**(2), 545 – 555 (2015)
25. Sfikas, G., Retsinas, G., Gatos, B.: Zoning aggregated hypercolumns for keyword spotting. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 283 – 288 (2016)
26. Sudholt, S., Fink, G.A.: PHOCNet: A deep convolutional neural network for word spotting in handwritten documents. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 277–282. Shenzhen, China (2016)
27. Sudholt, S., Fink, G.A.: Attribute CNNs for word spotting in handwritten documents. *Int. Journal on Document Analysis and Recognition* **21**(3), 199–218 (2018)
28. Toselli, A.H., Romero, V., Vidal, E., Sánchez, J.A.: Making two vast historical manuscript collections searchable and extracting meaningful textual features through large-scale probabilistic indexing. In: *Proc. Int. Conf. on Document Analysis and Recognition*. pp. 108–113. Sydney, NSW, Australia (2019)
29. Triguero, I., García, S., Herrera, F.: Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information Systems* **42**(2), 245–284 (Feb 2015)
30. Vats, E., Hast, A., Fornés, A.: Training-free and segmentation-free word spotting using feature matching and query expansion. In: *Proc. Int. Conf. on Document Analysis and Recognition*. pp. 1294–1299. Sydney, NSW, Australia (2019)
31. Vidal, E., Toselli, A.H., Puigcerver, J.: High performance query-by-example keyword spotting using query-by-string techniques. In: *Proc. Int. Conf. on Document Analysis and Recognition*. pp. 741–745. Nancy, France (2015)
32. Wilkinson, T., Brun, A.: Semantic and verbatim word spotting using deep neural networks. In: *Proc. Int. Conf. on Frontiers in Handwriting Recognition*. pp. 307 – 312 (2016)
33. Wilkinson, T., Lindström, J., Brun, A.: Neural ctrl-f: Segmentation-free query-by-string word spotting in handwritten manuscript collections. In: *Proc. Int. Conf. on Computer Vision*. pp. 4443–4452. Venice, Italy (2017)
34. Wolf, F., Oberdiek, P., Fink, G.A.: Exploring confidence measures for word spotting in heterogeneous datasets. In: *Proc. Int. Conf. on Document Analysis and Recognition*. pp. 583–588. Sydney, NSW, Australia (2019)
35. Zagoris, K., Pratikakis, I., Gatos, B.: Unsupervised word spotting in historical handwritten document images using document-oriented local features. *IEEE Trans. on Image Processing* **26**(8), 4032–4041 (Aug 2017)