2009 10th International Conference on Document Analysis and Recognition

A New Framework for Recognition of Heavily Degraded Characters in Historical Typewritten Documents Based on Semi-Supervised Clustering[†]

S. Pletschacher[1], J. Hu[2] and A. Antonacopoulos[1]

[1]*Pattern Recognition and Image Analysis (PRImA) Research Lab*
School of Computing, Science and Engineering, University of Salford, Greater Manchester, United Kingdom
http://www.primaresearch.org

[2]*IBM T.J. Watson Research Center*
1101 Kitchawan Road, Route 134 Yorktown Heights, NY 10598
jyhu@us.ibm.com

Abstract

This paper presents a new semi-supervised clustering framework to the recognition of heavily degraded characters in historical typewritten documents, where off-the-shelf OCR typically fails. The constraints are generated using typographical (collection-independent) domain knowledge and are used to guide both sample (glyph set) partitioning and metric learning. Experimental results using simple features provide encouraging evidence that this approach can lead to significantly improved clustering results compared to simple K-Means clustering, as well as to clustering using a state-of-the art OCR engine.

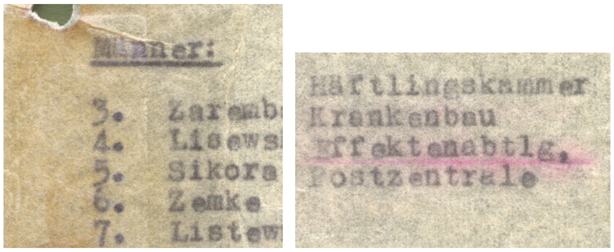
1 Introduction

There is a considerable public, historical as well as political interest in the analysis of large collections of administrative documents of the 20th century and their conversion into digital archives and libraries.

The majority of office documents and official correspondence of the 20th century are *typewritten*, a fact that introduces certain unique challenges to their recognition. First, in contrast to other printed documents, each individual glyph (character) within a document may appear considerably stronger or more faint than its neighbours. This is in direct relation to both the amount of force used when pressing the corresponding key and to the condition of the actual striking head of the particular key.

Second, many typewritten documents survive only as carbon copies of the originals, produced on a very thin paper (a.k.a. Japanese paper) which has prominent texture. Due to the mechanical nature of the typing process (the force from the typewriter key has to be transferred through the original paper and through the carbon sheet before a character is produced on the carbon copy) the characters on the carbon copy are usually blurred.

Historical typewritten documents are also affected by problems of ageing and repeated use, manifesting themselves as discolouration, disintegration of document parts, stains, punch holes, tears, rust from paperclips etc. Examples of scanned carbon-copy historical typewritten material is shown in Figure 1.



Männer:
3. Zaremb
4. Lisews
5. Sikora
6. Zemke
7. Listew

Häftlingskammer
Krankenbau
Effektenabtlg.
Postzentrale

Fig 1. Examples of typewritten material used.

As perhaps expected, state-of-the-art commercial OCR systems fail to recognise the majority of the characters in this document class (this empirical statement is experimentally validated below). The main reasons are, the presence of background texture, faint characters that appear broken and blurred characters that are filled-in and/or touching with others. These are acknowledged challenges for any OCR system.

There have been remarkably few reports in the literature related to the analysis of typewritten documents. Most approaches focus on pre-OCR enhancement of degraded typewritten characters. Cannon *et al.* [1] attempt to enhance only bilevel images and address artefacts that are of different nature than those found on degraded historical documents. In terms of challenging historical documents, Antonacopoulos and Karatzas [2] presented a study of the effects of different binarisation techniques at different segmentation levels while Antonacopoulos and Casado Castilla [3] proposed a new text recovery approach for typewritten documents. Both of those approaches relied on commercial systems for character recognition. An exception is the experimental word-level approach developed based on the VIADOCS typewritten

[†] This work has been supported in part through the EU 7th Framework Programme grant IMPACT (Ref: 215064).

978-0-7695-3725-2/09 $25.00 © 2009 IEEE
DOI 10.1109/ICDAR.2009.267

index cards [4] which used special knowledge of the particular subject area (natural history taxonomies) and of the specific archival organisation of the index cards to generate candidates for recognition of word images.

This paper proposes a new framework for *recognizing* particularly challenging collections of historical documents, such as typewritten documents of Word War II [5]. In addition to suffering from severe degradations (as mentioned earlier), such documents contain text (e.g. surnames with variable spellings) that can rarely be found in dictionaries, much less belong to a closed vocabulary set.

The key concept of the new framework is the combination of collection-independent domain knowledge (such as typography conventions) with human feedback in an iterative manner to gradually refine the system's understanding of the unique characteristics of the specific document collections, finally leading to a collection-dependent OCR engine.

The main focus of this paper is the initial semi-supervised clustering stage where a full partition of the samples (glyphs) is generated from a completely unlabeled set of training glyph samples.

The general concept, background and stages of the proposed framework are examined in more detail in Section 2. The experimental process and corresponding results are presented and discussed in Section 3, while the general remarks of Section 4 conclude the paper.

2 Iterative approach based on semi-supervised clustering

The system starts with a set of completely unlabeled training samples extracted from the collection, and clusters the samples in a semi-supervised setting. Constraints derived from largely collection-independent algorithms are used to guide not only the partition of the sample space, but also the learning of collection-specific metrics. These clusters and metrics then serve as starting points in an iterative process, where at each step human feed back is used to generate new constraints, which are in turn used to modify cluster memberships as well as the metrics. At the end of this process, the system will not only have learned "pure" clusters from the training samples, but also appropriate metrics, both of which can then be used to create (train) a domain-specific OCR engine.

2.1 Background

Semi-supervised clustering refers to a group of methodologies that use incomplete class labels or pair-wise constraints on data samples to aid unsupervised clustering. It is the focus of many recent studies as it provides alternative ways to learn from large amounts of unlabeled data combined with limited labelled data. It also offers a convenient framework for incorporating potentially incomplete domain knowledge in data exploration.

The semi-supervision is typically provided in one of two forms, as class labels (seeds) [6,8], or pair-wise *must-link (ML)* or *cannot-link (CL)* constraints [7,10]. Earlier approaches were purely constraint-based, where the labels or pair-wise constraints are used to guide the algorithm towards a partition that is most consistent with the given constraints [6,10]. Metric learning from constraints were later introduced, e.g. [11]. More recently, Bilenko et. al. presented an integrated constraints and metric learning approach, where the constraints are used to adjust both clustering assignments and metrics [7].

While there has been a large body of work on the design of semi-supervised clustering algorithms, relatively little study has been carried out on different constraint generation methodologies and their effectiveness in different applications [8,10]. Intuitively, for the constraints to help, they should encode information that is not readily extractable from the basic features themselves. Such "side" information could come from human input (through limited manual labelling), or domain knowledge.

2.2 Features

Typewritten document images from World War II archives [5] are first binarised. In order to be able to make a baseline comparison with other approaches, no restoration [3] is attempted. The documents are then segmented down to glyph level.

The glyphs are fed into a feature extraction method (new extension of [9]) that calculates expressive values to be used as input to the semi-supervised clustering. The result is one feature vector for each glyph which represents a number of geometrical characteristics, density information, and localised features. Feature values currently used for glyphs are: *width* (normalised), *height* (normalised), *width to height ratio*, *number of black pixels* (normalised), *number of white pixels* (normalised), *black-to-white ratio*, as well as the individual *numbers of black pixels* and *black pixel densities* in each of the nine rectangles resulting from a regular 3×3 partition of a glyph.

It should be noted that, in most cases, structural features of printed characters cannot be used reliably due to the excessive amount of noisy pixels present e.g. strokes are wrongly connected or broken.

2.3 Automated constraint generation

Constraints need to be generated to inform the semi-supervised clustering algorithm. It is crucial that constraints used in the learning process introduce as few errors as possible. In this particular case, domain knowledge (related to typography) is used to automatically generate reliable constraints. (i.e. identifying easily separable features related to typical glyph characteristics).

The rest of this section explains how *cannot-link* and *must-link* constraints are generated using specific reliable features to provide input to the semi-supervised clustering where all features mentioned in Section 2.2 are taken into account. Figure 2 illustrates the process.

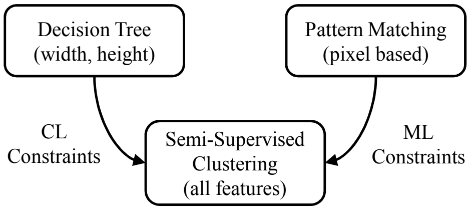


Fig. 2. Automated generation of CL and ML constraints for semi-supervised glyph clustering.

2.3.1. Cannot-link (CL) constraints. The basic requirement on this type of constraints is that two glyphs marked as “cannot-link” have very high probability of belonging to different classes in the ground-truth. To establish CL constraints a relatively small number of super-clusters can be produced which do not have to be pure with regard to the true glyph classes they contain. However, the intersection of the true classes between super-clusters has to be small (the sets should ideally be disjoint). If, for example, super-cluster A contains “i”s and “l”s then a super-cluster B may contain “m”s and “w”s but should ideally not contain any single “i” or “l”.

For our initial evaluation we used CL constraints based on the domain knowledge that characters with very different aspect ratios typically belong to different character classes. The constraints are calculated using a decision tree, operating only on glyph width and height. Glyphs in typewritten documents exhibit a typical characteristic graph for both width and height. While there is a linear middle section in each graph (corresponding to glyphs which can be hardly distinguished using width and height) the two ranges on the left and right ends of the graph exhibit higher slope resulting from glyphs that are significantly different from the rest. Optimal thresholds (used by the decision tree) are set accordingly, to exclude glyphs of the linear part from the resulting super-clusters as they are likely to cause false CL constraints.

This pre-clustering process produces four distinct super-clusters and one cluster containing all uncertain glyphs: (*i*) *narrow and short* (e.g. “.”), (*ii*) *narrow and tall* (e.g. “l”), (*iii*) *wide and short* (e.g. “w”), (*iv*) *wide and tall* (e.g. “H”), and (*v*) *otherwise* (remaining cases where a reliable decision cannot be made). Pairs of glyphs between the different super-clusters (apart from cluster *v*) form the required CL pairs.

2.3.2. Must-link (ML) constraints. This second type of constraints is used to signify glyphs of the *same true class*. The goal is therefore to find glyphs which definitely represent the same character or, in other words, to find absolutely pure clusters. This process does not necessarily have to be extensive, as only a reasonable set of ML constraints is required.

Automatic ML constraint generation is achieved by applying a very strict pre-clustering. The current method is based on vector quantisation with pixel-level pattern matching of glyph images as a similarity measure. Safe recognition of must-link candidates is possible with an allowed error of less than 10% of the total number of pixels in a glyph. This way, only the visually most similar glyphs are grouped together (see Fig. 3) and the number of clusters is typically larger than the number of actual classes (some of them have to be merged in the course of the subsequent semi-supervised clustering).

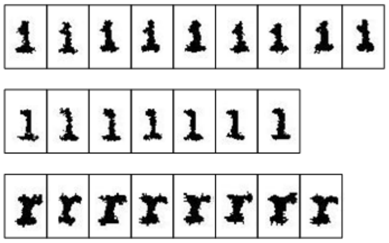


Fig. 3. Pre-clustering example result – clusters of similar glyphs then form ML constraints.

2.4 Semi-supervised clustering approach

The ML and CL constraints are incorporated into an iterative clustering procedure using The Semi-supervised Clustering with Metric Learning (MPCK-Means) algorithm developed by Bilenko et. al. [7]. MPCK-Means is particularly attractive for our application for several reasons. First, it is a K-Means based algorithm and as such is highly scalable. Second, it adapts to the constraints through both cluster-assignments and metric learning. The latter aspect is important because different document collections are expected to have different optimal metrics. Finally, it is able to learn individual metrics for each cluster, allowing clusters of different shapes. This is desirable because features that are most discriminative for some characters could be significantly different than those for others.

In MPCK-means, the Euclidean distance is parameterized using a symmetric positive-definite matrix A_{l_i} as follows:

$$\| x_i - \mu_{l_i} \|_{A_{l_i}} = \sqrt{(x_i - \mu_{l_i})^T A_{l_i} (x_i - \mu_{l_i})}$$

where A_{l_i} is the weight matrix for cluster l_i, and μ_{l_i} is the centroid of cluster l_i.

The objective function that combines metric learning with pair-wise constraints is defined as:

$$\begin{aligned}\Im = &\sum_{x_i} (\| x_i - \mu_{l_i} \|^2_{A_{l_i}} - \log(\det(A_{l_i})) \\ &+ \sum_{(x_i,x_j)\in M} w_{ij} F_M(x_i,x_j) I[l_i \neq l_j] \\ &+ \sum_{(x_i,x_j)\in C} \overline{w}_{ij} F_C(x_i,x_j) I[l_i = l_j].\end{aligned}$$

Here M is the set of must-link (ML) constraints, C is the set of cannot-link (CL) constraints, I is the indicator function, $F_M(x_i,x_j)$ and $F_C(x_i,x_j)$ are constraint violation penalty terms defined using the distance between x_i and x_j, w_{ij} is the weight for ML penalty and $\overline{w}_{ij}$ is the weight for CL penalty.

The algorithm locally minimizes the objective function by iterating through estimation (E) and maximization (M) steps. Constraints are utilized when assigning points to clusters, and the distance metric is adapted by re-estimating the weight matrices A_{l_i} during each iteration based on the current cluster assignments and constraint violations. For more details see [6].

3 Experiments

Experiments were carried out using glyphs segmented from binarised historical typewritten documents (see Section 2.2). The dataset contains 643 samples belonging to 55 character classes. For validation purpose all samples were labelled using a semi-automated tool [9] which identifies glyph candidates on a page and also suggests groups of similar glyphs for more efficient labelling. A set of 11 simple features such as width, height and percent of black pixels in 3x3 regions (see Section 2.2), and a fixed number of 55 clusters were used in all clustering experiments.

The main objective of the experiments is to investigate the benefit of semi-supervised clustering under two inherent challenges of this particular application: 1) limited and potentially noisy constraints due to the poor quality of the original samples and 2) large number of clusters.

In order to study the effect of the number of constraints and the trade-off between the number and quality of the constraints, we first generated the ML and CL sets using the automatically selected thresholds as described in Section 2.3, then varied the thresholds to generate sets with more or fewer constraints. The CL and ML constraints were evaluated separately, since they are generated through very different mechanisms and are expected to have different effects on clustering results.

Each constraint set was fed to the MPCK-means algorithm with the default constraint weight of 1.0. Clustering results are evaluated using the widely used F-measure ([7]). For comparison, we also generated clustering results using K-means and a state-of-art OCR engine, ABBYY FineReader 9, which in effect produces labelled clusters i.e. the recognised characters.

The F-measures for these baseline methods are *0.581* and *0.583*, respectively.

Table 1. Clustering Performance with ML constraints.

	# ML assignments	# ML constraints	# wrong constraints	F-measure
ML1	83	336	0	0.59
ML2	128	406	4	0.632
ML3	**206**	**731**	**12**	**0.701**
ML4	268	981	33	0.701
ML5	316	1566	153	0.693
ML6	372	2062	180	0.678

Table 1 shows the results of applying ML constraint sets of varying sizes and accuracy, with ML3 (in bold) being the set generated using the automatically selected thresholds. As can be seen, the use of ML constraints leads to dramatic performance improvements over simple k-means as well as FineReader 9. The amount of improvement increases initially as the number of constraints increases. This trend continues even after errors are introduced into the constraints with more relaxed thresholds. As expected, the performance eventually starts to drop as the negative effect of the wrong constraints outweighs the positive effect of the correct ones. We note that the "operating range" for ML constraints is quite large - significant performance improvements can be observed for most of the sets. This is reassuring as it indicates that the system does not rely on a precisely determined set of thresholds in constraint generation.

Table 2 shows the results from similar experiments for CL constraints, with CL3 (in bold) being the set generated with the thresholds described in Section 2.3.1. The general movement of the performance with regard to the size and accuracy of the constraint sets is similar to that observed for ML constraints. However, the CL constraints in general do not improve the performance much. In fact, there is an initial dip in the performance, and it only pulls ahead of the baseline for three of the constraint sets in the middle, before dropping again after a small increase in the number of wrong constraints. We conjecture that one possible reason for this behaviour could be that the current metric-learning formulation in the MPCK-means algorithm may not be well suited for the situation where there are a large number of clusters.

Table 2. Clustering Performance with CL constraints.

	# CL assignments	# CL constraints	# wrong constraints	F-measure
CL1	56	443	0	0.565
CL2	84	1931	1	0.578
CL3	**86**	**2091**	**1**	**0.59**
CL4	127	5402	5	0.601
CL5	133	5912	7	0.611
CL6	137	6332	9	0.567

Finally, we ran a set of experiments on the combined effect of ML and CL constraints. While many ML/CL combinations could be made using the set of constraints listed in Table 1 and Table 2, for this initial study we simply used 6 sets generated using the most straightforward manner, by combining the ML and CL sets in order. Since ML clearly leads to more performance improvements than CL constraints in the separate experiments, we skewed the combined set by using 1.0 as ML weights, and 0.1 as CL weights. The results compared to the baselines are shown in Figure 4.

4 Concluding remarks

We have proposed a new framework for recognizing challenging collections of historical documents, and presented the implementation and evaluation of the first step within this framework - constraint based clustering of glyphs. The constraints are generated using collection-independent domain knowledge, and are used to guide both sample partitioning and metric learning. Preliminary experiments using simple features provide encouraging evidence that this approach can lead to significantly improved clustering results compared to simple K-Means clustering, as well as clustering using a state-of-the art generic engine.

Future work includes more experiments with larger test sets and more document types, and improvement of the metric learning formulation to enhance its robustness when faced with large number of clusters and noisy CL constraints. We would also like to explore other ways of generating constraints. For example, one possible source of ML constraints is to generate synthetic samples from the current ML labelled set using character distortion models, to further enrich the set of ML constraints. For some document collections, more CL constraints could potentially be generated by making use of language models. Finally, after the initial semi-supervised clustering, how to present the results to most effectively solicit human feedback to further refine the clusters remains a challenging open research problem.

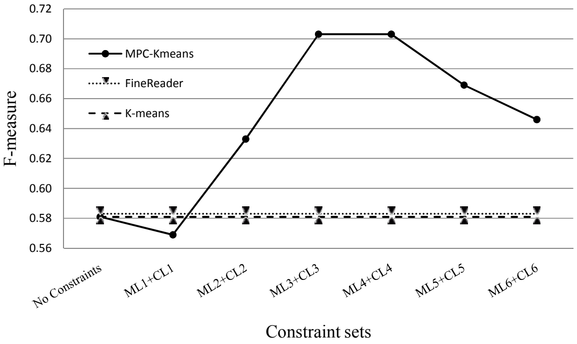


Fig. 4. Clustering performance with combined ML and CL constrains.

References

[1] M. Cannon, J. Hochberg and P. Kelly. "QUARC: A Remarkably Effective Method for Increasing the OCR Accuracy of Degraded Typewritten Documents", *Proc. 1999 Symp. on Document Image Understanding Technology (SDIUT'99)*, Annapolis, MD, May 1999, pp. 154-158.

[2] A. Antonacopoulos and D. Karatzas, "Semantics-Based Content Extraction in Typewritten Historical Documents", *Proc. 8th Int. Conf. on Document Analysis and Recognition (ICDAR2005)*, Seoul, South Korea, 2005, pp. 48–53.

[3] A. Antonacopoulos and C. Casado Castilla, "Flexible Text Recovery from Degraded Typewritten Historical Documents", *Proc. 18th Int. Conf. on Pattern Recognition (ICPR2006)*, Hong Kong, 2006, pp. 1062-1065.

[4] S. M. Lucas, G. Patoulas and A. C. Downton, "Fast lexicon-based word recognition in noisy images", *Proc. 7th Int. Conf. on Document Analysis and Recognition (ICDAR2003)*, Edinburgh, 2003, pp. 462-466.

[5] A. Antonacopoulos and D. Karatzas, "A Complete Approach to the Conversion of Typewritten Historical Documents for Digital Archives", in Document Analysis Systems VI, Springer LNCS 3163, 2004, pp. 90–101.

[6] S. Basu, A. Benerjee and R. Mooney, "Semi-supervised clustering by seeding", Proc. 19th ICML, Sydney, 2002.

[7] M. Bilenko, S. Basu and R.J. Mooney, "Integrating constraints and metric learning in semi-supervised clustering", *Proc. 21st ICML*, Banff, Canada, 2004.

[8] J. Hu, M. Singh and A. Mojsilovic, "Categorization Using Semi-Supervised Clustering", *Proc. 19th ICPR*, Tampa, Florida, 2008.

[9] S. Pletschacher, "Representation of Digitized Documents Using Document Specific Alphabets and Fonts, Soc. for Imaging Science and Technology (IS&T) Archiving 2008. Bern, Switzerland, 2008, pp. 198-202.

[10] K. Wagstaff, C. Cardie, S. Rogers and S. Schroedl, "Constrained K-means Clustering with Background Knowledge", Proc. 18th ICML, Massachusetts, US, 2001.

[11] E.P. Xing, A.Y. Ng, M.I. Jordan and S. Russell, "Distance metric learning with application to clustering with side information", Proc. 17th NIPS, Vancouver, Canada, 2003.