READ

RECOGNITION & ENRICHMENT
OF ARCHIVAL DOCUMENTS

D7.19

Model for Semi- and Unsupervised HTR Training P1

Approaches, Scenarios and first Results

Tobias Grüning, Gundram Leifert, Tobias Strauß, Roger Labahn
URO

Distribution: http://read.transkribus.eu/

READ
H2020 Project 674943

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 674943

Project ref no.	H2020 674943
Project acronym	READ
Project full title	Recognition and Enrichment of Archival Documents
Instrument	H2020-EINFRA-2015-1
Thematic priority	EINFRA-9-2015 - e-Infrastructures for virtual research environments (VRE)
Start date/duration	01 January 2016 / 42 Months

Distribution	Public
Contract. date of delivery	31.12.2016
Actual date of delivery	30.12.2016
Date of last update	21.12.2016
Deliverable number	D7.19
Deliverable title	Model for Semi- and Unsupervised HTR Training P1
Type	Demonstrator
Status & version	Final
Contributing WP(s)	WP7
Responsible beneficiary	URO
Other contributors	UPVLC
Internal reviewers	Joan Andreu Sánchez (UPVLC), Giorgos Sfikas (NCSR)
Author(s)	Tobias Grüning, Gundram Leifert, Tobias Strauß, Roger Labahn
EC project officer	Martin Majek
Keywords	semisupervised, unsupervised, alignment

Contents

Executive Summary

The first years deliverable describes the task and generic sub-tasks of it. First approaches and results are depicted and shortly explained. Regarding the scenarios of given transcripts with linebreaks and of a given language model first experiments were performed with promising results.

1 Introduction

The technologies for HTR recognition developed and used within READ rely on training (D7.1 & D7.7). Since, (basically) the more training data available, the better the HTR accuracy is, it is meaningful to provide as much training data as possible to the HTR system. The classical way to produce training data (for a line based HTR system, which is state-of-the-art up to now) is to create a set of line images and corresponding textual transcripts. This is expensive because a lot of user interaction is necessary.

For many documents the transcripts are already available. However, these transcripts are not assigned to line images. Therefore, it is *highly* motivated to assign those line images to transcripts – Task 7.7 targets this.

1.1 Task 7.7 - Semisupervised HTR Training

The following is copied from the Grant Agreement:

The semisupervised training process outlined in Sec.1.4.2 will be implemented and developed in this task. This process relies on computing reliable confidence measures at the word level and this tasks includes work to experiment with existing technologies and select those which result most appropriate. The training process will be initialized with Character Optical Model and Language Models trained in Tasks 7.1, 7.3 and 7.4. These models are then used to automatically transcribe new text line images (obtaining word segmentation as a by-product). This kind of unsupervised training is obviously much less powerful than the standard fully supervised training. Yet, it is still expected to lead to significant HTR accuracy improvements without incurring the prohibitive human transcription costs entailed by the conventional, supervised training workflow. An intermediate form of semisupervised training will be also implemented to take advantage of existing transcripts which may not be aligned with the physical lines of the text images. To this end, this task includes work to test, adapt and/or develop the text-image alignment techniques outlined in Sec.1.4.1. These techniques will make use of the developments and models produced in Task 7.1 or 7.3.

1.2 Generic Sub-Task of Task 7.7

Within this deliverable, Task 7.7 is divided into sub-tasks depending on the data and/or methods that are used for producing the additional semisupervised training data or training the models:

(a) images with corresponding transcripts (Task 7.7a). This (line) images can have additional information. In the following we concentrate on:

- transcripts with correct line breaks
- transcripts with correct page breaks
- transcripts without usable breaks

Remark 1. All transcripts available in this scenarios are *not* aligned. There is no correspondence given between the textual transcripts and physical positions in the image which is mandatory for the training process.

(b) images without transcripts (Task 7.7b)

- with language model
- without language model

Remark 2. Since for the URO HTR system it is an obvious first step to utilize available transcripts to improve the HTR accuracy, UROs work of the first year within READ focused on Task 7.7a. Task 7.7b was tackled by UPVLC and will be of major interest for URO in the next years.

2 Task 7.7a

Task 7.7a aims at aligning given transcripts to images and we therefore refer to it by *Text2Image*. Fig. 1 show an example of this alignment problem which is a real-world problem of the MoU Partner "University Rostock – Barlach Project"[1]. Since transcripts are produced in various ways under various conditions, there are scenarios where correct line- or page breaks are available, in others no breaks are usable at all. This leads to problems of different complexity.

Another circumstance is the availability of so-called *baselines* (see Definition 3). These baselines should underline the mainbody of the text on pages. They are either created manually or automatically by a *Layout Analysis process* (see Sec.1.4.9), but in both cases the baselines can be incorrect. This leads to additional problems in assigning transcripts to baselines.

Another issue is the order of the baselines because there is not always a natural order of them. For example, the baselines of marginalia can be placed between the associated paragraph text or after it.

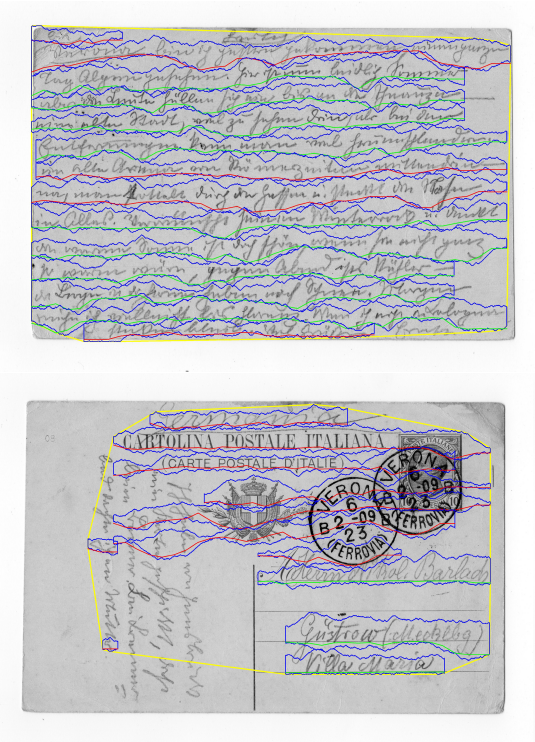
2.1 General Assumptions

We may assume to have a sorted set of baselines – either manually or automatically produced:

Definition 3 (Baseline). A baseline

$$b = \Big((x_1, y_1), \ldots, (x_k, y_k) \Big)$$

[1] http://www.germanistik.uni-rostock.de/forschung/ernst-barlach-briefedition/



Verona bin ich gestern gekommen einen ganzen
Tag Alpen gesehen. Hir ist nun leidlich Sommer
abr die Leute hüllen sich ein bis an die Schnauze
eine alte Stadt, viel zu sehen drin, abr bei den
Entfernungen kann man mal herumschlendern.
in Alles. Verwünscht seinen Winterrock u. denkt
die warme Sonne ist doch schön, wenn sie nicht ganz
so warm wäre, gegen Abend ists kühler –
de Berge in dr Ferne haben noch Schnee. Morgen
reise ich vielleicht bis Florenz. Wen ich nicht in Bologna
steken bleib. Viel Grüße Ernst
eine Karte geschickt, dicht
beim Brenner. Bei Brenner-
bad dachte ich an Mutter.

Germania

Cartolina Postale Italiana
An Herrn Nikol. Barlach
Güstrow (Mecklbg)
Villa Maria

Figure 1: *The Text2Image task with correct line breaks:* For one or more given images (with baselines and surrounding polygons) and a text file, the tool tries to find the corresponding transcript for each baseline. Found matches are emphasized by a green baseline (see Fig. 2).

of length $k \geq 2$ is a sorted list of points in an image. This line should be located under the text and has the direction from left to right.

Let $\mathcal{B}$ be the set of all possible baselines. For each page, we have a manually or automatically calculated list of baselines $B = (b_1, \ldots, b_n) \in \mathcal{B}^n, n \in \mathbb{N}$, whereas we always have to deal with correctly and incorrectly ordered lines.

In the same manner we have a set $\mathcal{T}$ of all possible transcripts and a list $T = (t_1, \ldots, t_m) \in \mathcal{T}^m, m \in \mathbb{N}$, of transcripts of a page in a natural order.

For given baseline $b \in \mathcal{B}$ and transcript $t \in \mathcal{T}$ we want to get a confidence that the alignment is correct.

Definition 4 (Cost function). Let $\mathcal{B}$ be the set of baselines and $\mathcal{T}$ the set of transcriptions. We call

$$c : \mathcal{B} \times \mathcal{T} \to \mathbb{R}^+$$

the *cost function*.

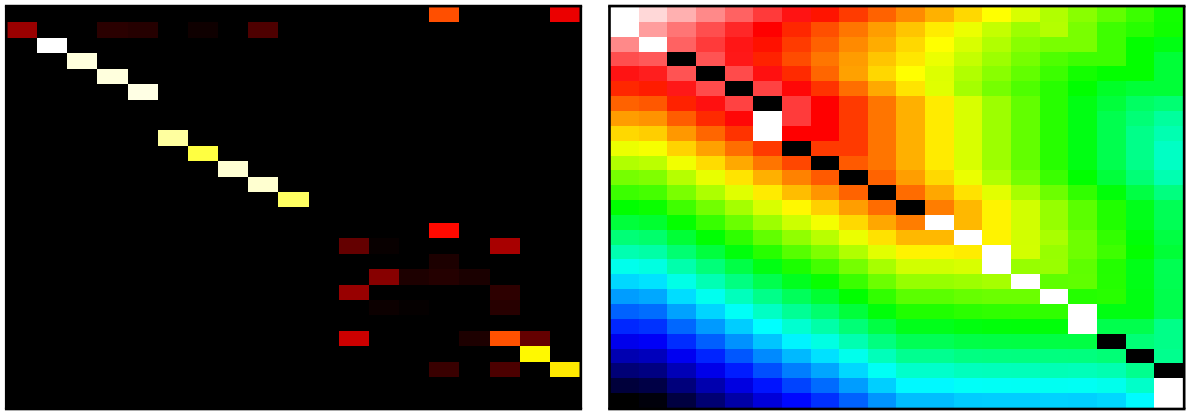
In the case of Neural Network based HTR (like used in [5]), *Connectionist Temporal Classification* can be used to calculate the probability that a given transcript is depicted in the image resulting from a given baseline [3]. The cost function has to be chosen such that a threshold $\tau \in \mathbb{R}^+$ determines whether or not a corresponding pair (b_i, t_j) will be saved as a training sample (e.g. see [6]). If so, we call the pair a *match*. The aim of all following algorithms is to match as much pairs as possible, whereby the number of false alignments should be kept minimal.

2.2 Transcripts with Correct Line Breaks

If the line breaks and the page breaks are given, one has a list $T := (t_1, \ldots, t_m)$ of transcriptions for a given page. With the list B of baselines one can calculate the match confidence for each pair (b_i, t_j). This leads to a match matrix $C \in (\mathbb{R}^+)^{n \times m}$ with $c_{i,j} := c(b_i, t_j)$ (see Fig. 2(a)). There are two general ways to find the best matches in C:

With Reading-Order: If the order of B and T can be assumed to be the same, dynamic programming through C can be used (see Fig. 2(b)). The idea is to successively match or delete baselines and transcriptions so that a defined cost function is minimal. Therefore, one has to define costs to ignore a baseline or a transcription. A first good choice is to set those costs to the threshold τ. Furthermore, also the matching costs should be bounded by τ. At the end, one receives an optimal alignment between both sequences, containing deletions and matchings with corresponding costs. All the matchings with costs lower than τ can be used as training samples.

Without Reading-Order: The other approach is to successively take out the pairs (b_i, t_j) with lowest costs $c_{i,j}$. If the cost is lower than τ the pair is taken as a training sample, otherwise the process stops and no further matches are calculated on the page. To avoid that baselines and transcriptions are matched more than once, specific



(a) Match matrix C (b) Dynamic programming matrix

Figure 2: *Matching of baselines to transcripts.* 2(a): The match matrix C shows the costs to match a baseline (row) to a transcription (column) shown in Fig. 1. White refers to low costs, while black indicates values exceeding a certain threshold τ. 2(b): Each value (i, j) of this map indicates the costs needed to match the first i baselines with the first j transcriptions. The cost-minimal path is emphasized by black-white colors, where black denotes a match.

matches have to be prohibited: If a match $c_{\tilde{i},\tilde{j}}$ was already chosen, the set of samples $\left\{(b_i, t_j)|i = \tilde{i} \vee j = \tilde{j}\right\}$ is removed and not available for further matchings any more.

2.3 Transcripts with Correct Page Breaks

If no line breaks are available, the alignment is much harder. In Section 2.2, only entities had to be mapped – namely one baseline to one transcript. Without line breaks there is solely one transcript and a list of baselines – so there is another matching problem. If the transcript is split into words one also has to split the baseline into words. But then one has to deal with so-called *over- and under-segmentation*. In addition, hyphenations have to be handled properly.

As this problem will be tackled later in the project, we will report on it in later deliverables.

2.4 Transcripts without Usable Breaks

Having, maybe, hundreds of pages and thousands of lines, the problem is to find an algorithm that scales properly.

As this problem will be tackled later in the project, we will report on it in later deliverables.

3 Task 7.7b

Task 7.7b aims at generating training samples by transcribing text lines using a pre-trained HTR system. By scoring these transcripts, only reliable samples are used for training.

Remark 5. The work outlined in this section was done by UPVLC.

3.1 With Language Model

Classical Handwritten Text Recognition (HTR) is usually carried out using Hidden Markov Models (HMM) and Language Models (LM) that is formulated as:

$$W = \underset{w}{\operatorname{argmax}}\ P(w|x) = \underset{w}{\operatorname{argmax}}\ P(x|w)P(w)$$

where W is the best transcript for the line image x among all possible transcripts w.

$P(x|w)$ represents the optical modelling that is approximated with HMM and $P(w)$ is the LM that is approximated with n-grams.

Unsupervised learning refers here to the automatic learning of $P(x|w)$ without human intervention. Some attempt for dealing with this problem has appeared recently [4], but it seems that these results can not been reproduced since many heuristics were necessary. The problem is that without an appropriate initialization of the HMM for computing $P(x|w)$, the EM algorithm used for training the HMM falls in a very poor local optimum and good results can not been achieved. Therefore, an alternative is to start with HMM initialized somehow. Preliminarily ideas were researched in [1], [2] and we have performed preliminary experiments in this direction. The idea in semisupervised learning is the following (in all cases the lines have to be previously detected):

1. Obtain initial HMM trained in a supervised way, with a very small amount of lines. We call this "HMM training with forced alignment".

2. Use the current HMM and a very good LM for obtaining transcripts automatically from a large amount of un-transcribed lines together with confidence measures (at character level or at word level).

3. Use the more confident transcripts (and the corresponding images) from step 2 to perform HMM training with forced alignment.

4. Go to step 2 until convergence.

It is expected in the previous process that initial HMM play as anchors that helps to fix position in sentences obtained from the LM to the line images. We have performed some very preliminarily experiments with this idea in the dataset described in [4] and the results are promising, but a more comprehensive research is currently under development. HMM in step 1 can be obtained with other techniques, like a pool of HMM, in order to remove the user completely from the production loop.

3.2 Without Language Model

As this problem will be tackled later in the project, we will report on it in later deliverables.

References

[1] Volkmar Frinken et al. "Co-training for handwritten word recognition". In: *2011 International Conference on Document Analysis and Recognition*. IEEE. 2011, pp. 314–318. DOI: 10.1109/ICDAR.2011.71.

[2] Volkmar Frinken et al. "Semi-supervised learning for cursive handwriting recognition using keyword spotting". In: *2012 International Conference on Frontiers in Handwriting Recognition*. IEEE. 2012, pp. 49–54. DOI: 10.1109/ICFHR.2012.268.

[3] A. Graves et al. "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Nets". In: *ICML '06: Proceedings of the International Conference on Machine Learning*. 2006. DOI: 10.1145/1143844.1143891.

[4] Michal Kozielski et al. "Towards Unsupervised Learning for Handwriting Recognition". In: *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*. IEEE. 2014, pp. 549–554. DOI: 10.1109/ICFHR.2014.98.

[5] Gundram Leifert et al. "Cells in Multidimensional Recurrent Neural Networks". In: *Journal of Machine Learning Research* 17.97 (2016), pp. 1–37. URL: http://jmlr.org/papers/v17/14-203.html.

[6] Tobias Strauß et al. "CITlab ARGUS for Keyword Search in Historical Handwritten Documents: Description of CITlab's System for the ImageCLEF 2016 Handwritten Scanned Document Retrieval Task". In: CEUR Workshop Proceedings. Évora, Portugal, Sept. 2016. URL: http://ceur-ws.org/Vol-1609/16090399.pdf.