



Account ▼

<https://readcoop.eu>[Transkribus \(/transkribus/\) >](#)[How-To Guides \(https://readcoop.eu/transkribus/resources/how-to-guides/\) >](#)

How To Train and Apply Handwritten Text Recognition Models in Transkribus eXpert

How To Train and Apply Handwritten Text Recognition Models in Transkribus eXpert

[🔍 Transkribus Tools](#) [📋 Transkribus Expert Client](#)*Last update 7 months ago*[« How-To's overview \(https://readcoop.eu/transkribus/resources/how-to-guides/\).](#)

► About Transkribus ()

Table of Contents



How To Train and Apply Handwritten Text Recognition Models in Transkribus eXpert

[Introduction](#)

Preparation

Training

Parameters

Max-nr. of Epochs

Early Stopping

Base Model

Learning Rate

Image Type

Use existing line polygons for training

Omit lines by tag

Reverse Text

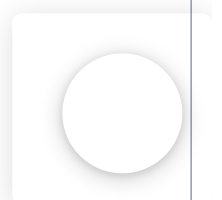
Training Tags

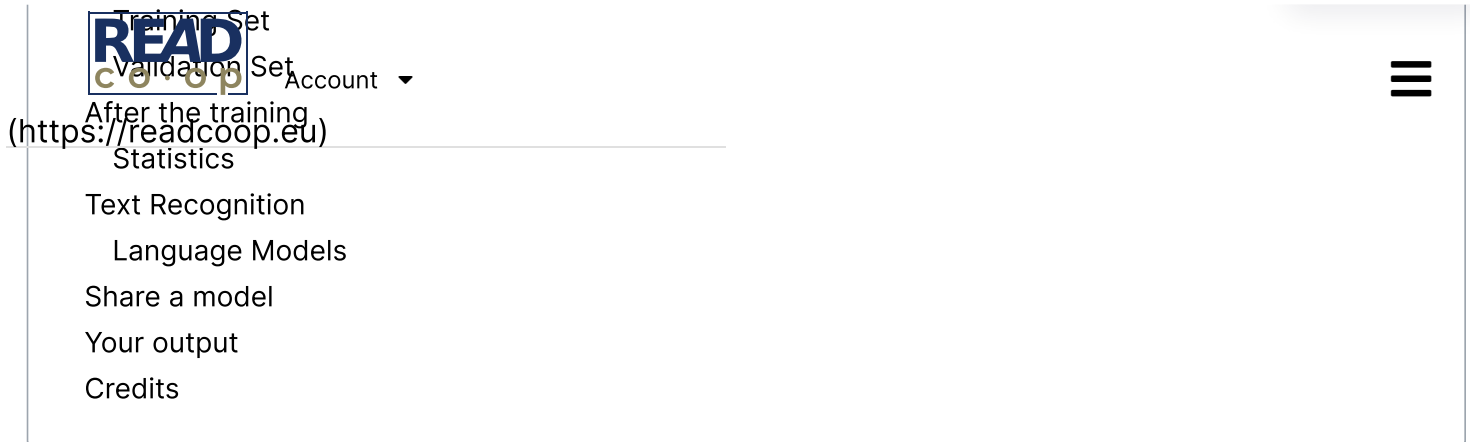
Advanced Parameters

Preprocessing

Model parameters

Training parameters





This guide explains how to use Transkribus eXpert to train a Handwritten Text Recognition (HTR) model to recognise your documents. After training, the model will help you to automatically transcribe and search your collection.

Introduction

- The Transkribus platform allows users to train a Handwritten Text Recognition (HTR) model to automatically process a collection of documents. The model needs to be trained to recognise a certain style of writing by being shown images of documents and their accurate transcriptions.
- For the training of a model between 5,000 and 15,000 words (around 25-75 pages) of transcribed material are required. If you are working with printed rather than handwritten text, a smaller amount of training data is usually required.
- With the use of a base model the amount of required training data can be reduced. As base model you can either use one of the publicly available models in Transkribus, if there is a suitable one for your documents or one of your own models, which you have already trained before. An overview of the currently available public models you can find here (<https://readcoop.eu/transkribus/public-models/>).

Preparation

- We recommend that you start the training process with between 5,000 and 15,000 words of transcribed material (25-75 pages), depending on if it is printed or handwritten text.
- The neural networks of the Handwritten Text Recognition engine learn quickly; the more training data they have, the better the results will be.

READ models can reduce the amount of training data required. As base model you can either use one of the publicly available models in Transkribus, if there is a suitable one (https://readcoop.eu/), or one of your documents or one of your models. Only PyLaia-models can be used as base models. You will find an overview of the currently available public models here (https://readcoop.eu/transkribus/howto/public-models-in-transkribus/).

- You can create training data for HTR in Transkribus by uploading images and transcribing text. For full instructions, see How To Transcribe Documents with Transkribus – Introduction (https://readcoop.eu/transkribus/howto/how-to-transcribe-documents-with-transkribus-introduction/).

Training

Once you have between 25 and 75 transcribed pages, it is time to train the Text Recognition model. Click on the **“Tools”** tab. Under the **“Model Training”** section, click on **“Train a new model”**.

The Model Training window pops up. By default, “PyLaia HTR”, the engine we are interested in, is selected, as shown in the figure below.

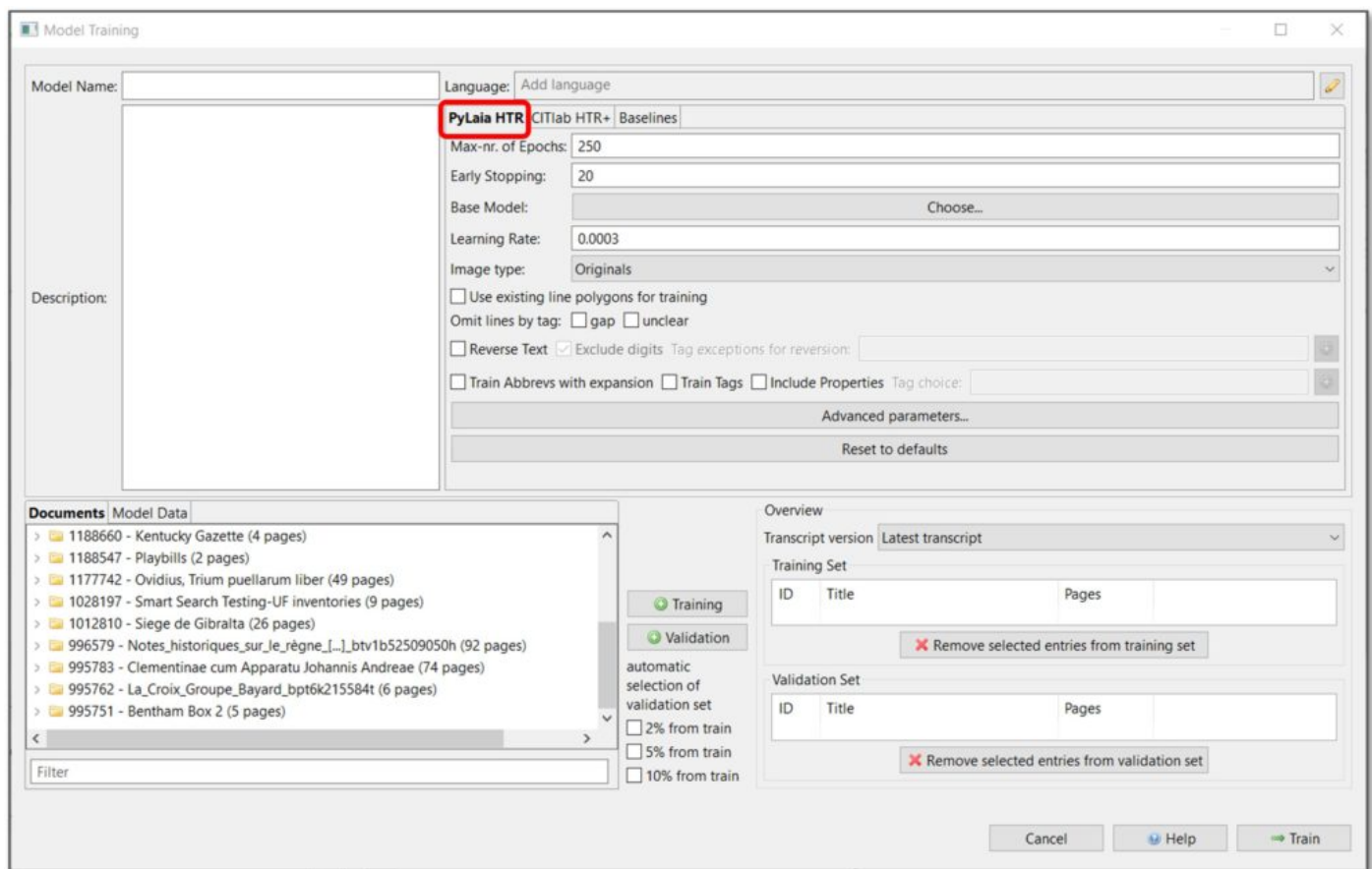


Figure 1. Model Training window

In the upper section, you will need to add details about your model:

Model Name (chosen by you)
 Language (of your documents)
 (https://readcoop.eu) Description (of your model and the documents on which it is trained)



Parameters

The parameters for PyLaia can be found in the upper right section of the window, under “Language”.

PyLaia HTR CITlab HTR+ Baselines

Max-nr. of Epochs: 250

Early Stopping: 20

Base Model: Choose...

Learning Rate: 0.0003

Image type: Originals

☐ Use existing line polygons for training

Omit lines by tag: ☐ gap ☐ unclear

☐ Reverse Text ☒ Exclude digits Tag exceptions for reversion:

☐ Train Abbrevs with expansion ☐ Train Tags ☐ Include Properties Tag choice:

Advanced parameters...

Figure 2. PyLaia HTR Training Parameters

In detail, they are:

Max-nr. of Epochs

The number of epochs refers to the number of times that the training data is evaluated. You can increase the maximum number of epochs, but be aware that the training process will take longer. Furthermore, note that the training will be stopped automatically when the model no longer improves (i.e. it has reached the lowest possible CER). For a start, it makes sense to stick to the default setting of 250.

Early Stopping

The value of 20 means that if, after 20 epochs, the CER of the Validation Set does not go down, the training will be stopped.

NOTE: Instant here and for trainings in general: the Validation Set needs to be variable and should possibly contain all types of elements of the documents included in the training set. If there is not or little variation in the Validation Set, the model may stop too early. (<https://readcoop.eu>)

Therefore if your validation set is rather small, please increase the “Early Stopping”-value in order to avoid the training from stopping before it has seen all the training data. Conclusion of this: don't save effort at the Validation Set.

Base Model

It is possible to add a base model to your training. If you choose this option, the neural nets will learn quicker and you will save time. To have a benefit, the base model needs to be similar to the writing it should recognise. With the help of a base model, it is possible to speed up the training process. Likely you will also improve the quality of your recognition results with a base model. However, this is not always guaranteed and has to be tested for the specific case.

One big benefit of working with base models is, that they make it possible to start with a smaller amount of training pages, which means that the transcription workload is reduced.

To use a base model, you simply need to choose the desired one with the “Choose...” button next to “Base Model”. You can select one of the PyLaia public models or a PyLaia model of yours.

Learning Rate

The “Learning Rate” defines the increment from one epoch to another, so how fast the training will proceed. With a higher value, the CER will go down faster. But, the higher the value, the higher the risk that details are overlooked.

This value is adaptive and will be adjusted automatically. The training is influenced though by the value it is started with. You can go with the default setting here.

Image Type

We have had some cases where the pre-processing took too much time. If this happens to you, you can switch the “Image Type” to “Compressed”.

You can proceed in the following way: start the training with “Original”. When the training has started (“Running” status), every now and then check the progress of the pre-processing with the “Jobs”-button. In case it gets stuck, you can cancel the job and restart it with the “Compressed”-setting.

Use existing line polygons for training



Account ▼



If you flag this option, line polygons will not be computed during the training (as it happens by default), but the existing ones will be used. It refers only to lines (<https://readcoop.eu/>) (<https://readcoop.eu/glossary/line-region/>), not to baselines.

During the recognition, then, similar line polygons should be used for the best performance of the resulting trained model.

Omit lines by tag

With this option, you can omit lines containing words tagged as “gap” and/or “unclear” from the training. Please note that the whole line will be ignored during the training, not only the unclear word: this happens because the training happens on the line level.

Reverse Text

Use this option to reverse text during the training when the writing direction in the image is opposite to that of the transcription, e.g. the text was written right-to-left and transcribed left-to-right. You can also decide to exclude digits or tagged text from reversion.

Training Tags

It is possible to train tags and properties if they are present in the ground truth, so that the model will automatically generate tags during the recognition. This feature works well with abbreviations and text styles and brings the best results for tags which are repeated in the same way (i.e. the same word) very often.

Select “Train Abbreviations with tags” to train the tagged abbreviations (“Abbrev” tag) and the respective “expansion” properties present in your ground truth.

For other tags, select the “Train Tags” option and click “Include Properties”; then use the green plus button to enter the list of tags that should be trained.

Advanced Parameters

In PyLaia, users can set several advanced parameters on their own. You can open the advanced parameters by clicking on the “Advanced parameters”-button at the bottom of the standard parameters.

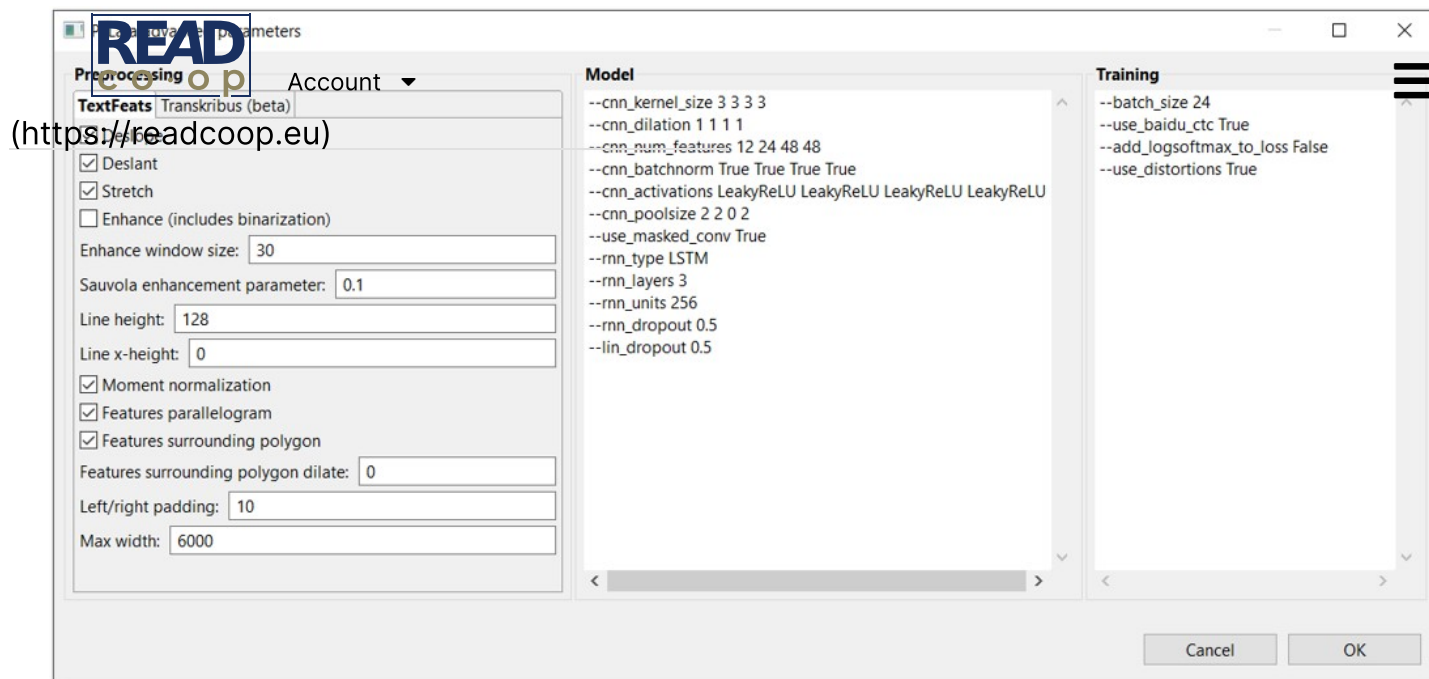


Figure 3. PyLaia Advanced Parameters

Preprocessing

Deslant: choose this option with cursive writing in order to straighten it. Leave out this option with printed documents, because if printed documents contain cursive passages in addition to the normal print characters, the effect can be upside down.

Deslope: allows more variation at the baselines, e.g. more tolerance at baselines, that are not exactly horizontally but slanting.



Stretch: this option is for narrow writing in order to uncompress it.

Enhance: that is a window, which goes over the baselines in order to optimize passages, which are difficult to read. This is useful if you have “noise” in the document.

Enhance window size: this setting refers to the option just explained and therefore only needs to be set, if you would like to use “Enhance”. This setting defines the size of the window.

Sauvola enhancement parameter: please stick to the default setting here.

Line height: value in pixels; if you need to increase the pixels of the images, you can do this here. 100 is a good value to go for. Attention: if the value is too high it might lead to an “out of memory order”. You can bypass this error in turn by lowering the value of the “batch size” (top left in the advanced parameters window), e.g. by half. Please be aware that the lower

this will make the training slower. The slow-down of the training relating to the batch size should be improved with the new version of PyLaia, which will set the batch size automatically.  

(<https://readcoop.eu>)

Line x-height: this setting applies to the descenders and ascenders. If you put this value, the „Line height” parameter will be ignored.

Please don't change the following parameters:

Moment normalization

Features parallelogram

Features surrounding polygon

Features surrounding polygon dilate

Left/right padding: 10 (default) means that 10 pixels will be added. This is useful if you are worried that parts of the line could be cut off.

Max width: maximum width that a line can reach; the rest will be cut off. 6000 (default) is already a high value. If you have huge pages, you can further increase this value.

Model parameters

For all those who are familiar with machine learning and the modification of neural nets. Therefore, these parameters are not further explained here.

Training parameters

Batch size: number of pages which are processed at once in the GPU. You can change this value by putting another number.

Use distortions True: the training set is artificially extended in order to increase the variation of the training set and in this way make the model more robust. If you are working on even writing and good scans, you don't need this option. To deactivate it, please write „False” instead of „True”.

The net structure of PyLaia can also be changed – a playground for people, who are familiar with machine learning. Modifications on the neural net can be done via the Github repository (<https://github.com/jpuigcerver/pylaia>).

(<https://readcoop.eu>) Next, you need to select the pages that you would like to be included in your set of training data.

To add all the pages of your document to the Training Set, click on the folder and click “+Training”. To add a specific sequence of pages from your document to the Training set, double-click on the folder, click on the first page you wish to include, hold down the “Shift” key on your keyboard and then click the last page. Then click “+Training”. To add individual pages from your document to the Training Set, double-click on the folder, hold down the “CTRL” key on your keyboard and select the pages you would like to use as training data. Then click “+Training”.

The pages you have selected will appear in the “Training Set” space.

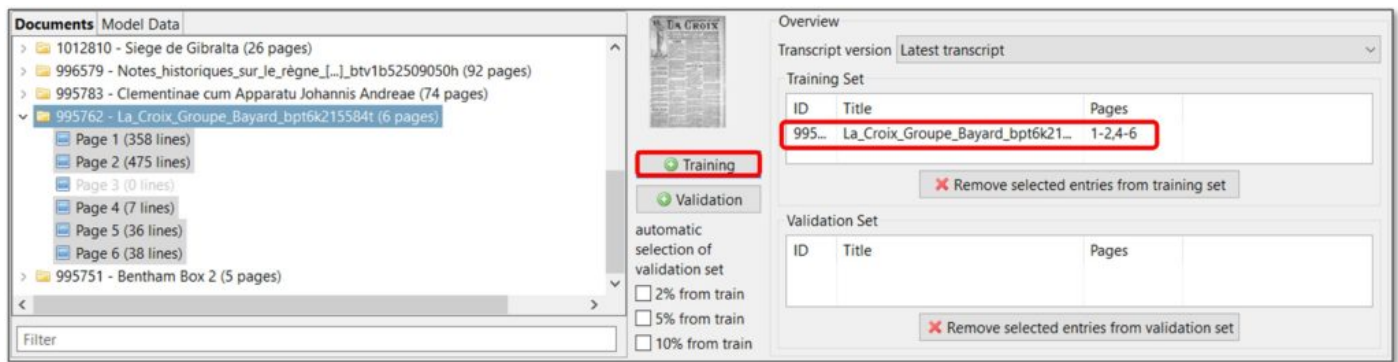


Figure 4. Training Set

Validation Set

During the training process, a Validation Set of pages is set aside and is not used to train the HTR. These test pages will then be used to assess the accuracy of your model.

We recommend that your Validation Set is about 10% of the Training Set. The pages in your Validation Set should be representative of the documents in your collection and comprise all the examples. To add pages to the Validation Set, follow the same process as above but click the “+Validation” button.

You can also automatically assign 2%, 5% or 10% of the Training Set to the Validation Set. Select the pages to add to the Training Set, flag the percentage you want to assign to the Validation Set and click the “+Training” button.

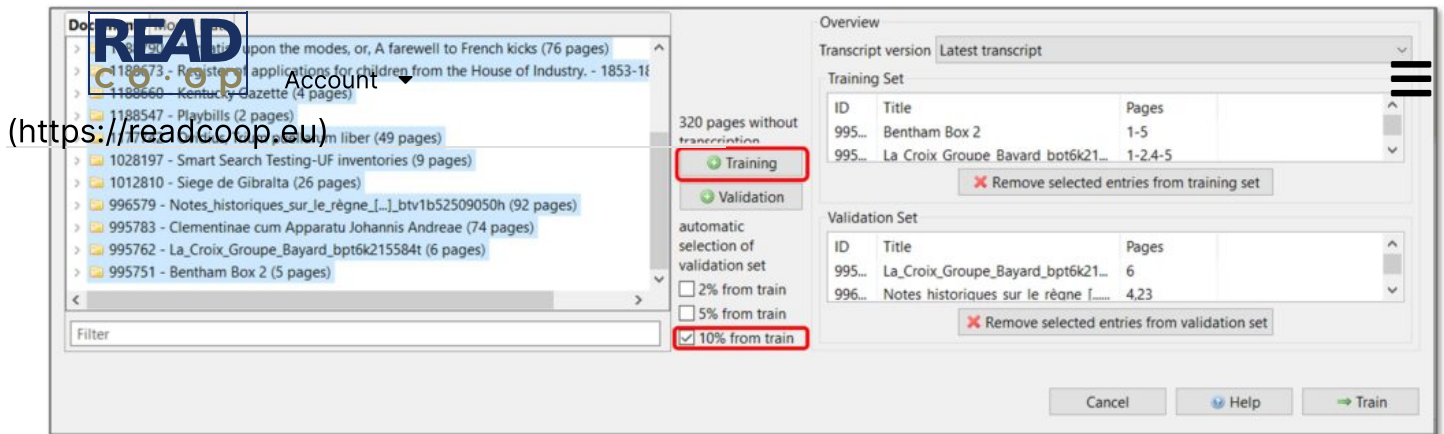


Figure 5. Automatic Selection Validation Set

To remove pages from the “Training Set” or “Validation Set”, click on the page and then click the red cross button.

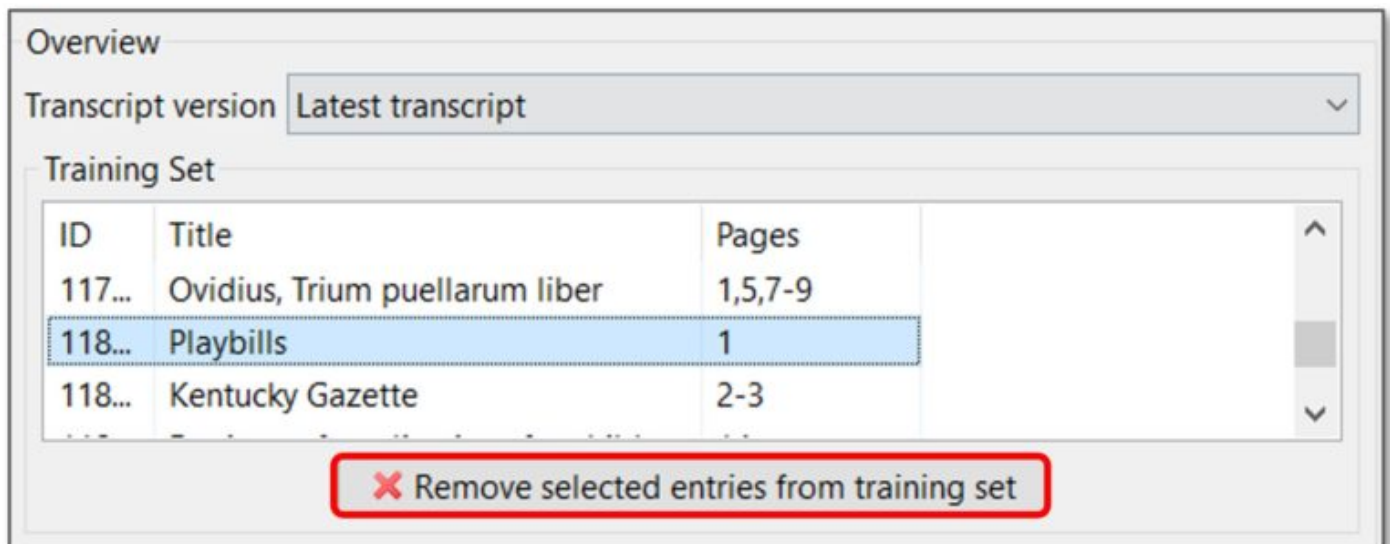


Figure 6. Remove pages from the Training/Validation Set

Then, start the training by clicking the “Train” button.

You can follow the progress of the training by clicking the “Jobs” button in the “Server” tab. The completion of every epoch will be shown in the Job’s description, and you will receive an email when the training process is completed.

Training a Text Recognition model takes from some hours to several days, depending on the server traffic. In the “Jobs” window, you can check your position in the queue (i.e. the number of trainings ahead of yours). You can perform other jobs in Transkribus or close the platform during the training process. If the Job status is “created” or “running”, please don’t start a new training, but just be patient and wait.

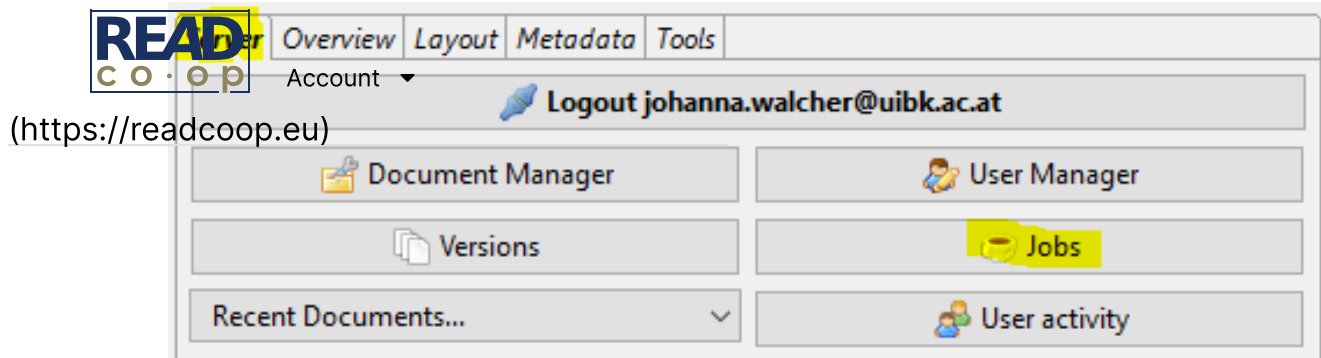


Figure 8. Check the progress of the training with the “Jobs” button

After the training

After the training of your model is finished, it will be available in your collection. In order to access it, click the “View models” button in the “Tools” tab.

The following window will open up, where you can see all the public models and the models you trained:

Choose a model

Public Models | text | PyLaia HTR | German (14)

1-14 / 14 | 1 1 |

Name	Language	Curator	Technol...	Creat...	nrO
Transkribus print 0.3	German; En...	guent...	PyLaia H...	27.08...	441
Transkribus German hand...	German	office...	PyLaia H...	16.08...	361
Generic Model 15th-16th ...	German	f.staud...	PyLaia H...	05.11...	792
Paul-Goldmann_German-K...	German	laura.u...	PyLaia H...	24.10...	633
NorFraktur_1600_PyLaia	Norwegian;...	yngvil...	PyLaia H...	07.04...	106
Transkribus Print M1	German; En...	b.anzi...	PyLaia H...	19.02...	506
German_Kurrent_XIX-XX...	German	tobias...	PyLaia H...	17.11...	634
Transkribus Early Kurrent ...	German	b.anzi...	PyLaia H...	26.10...	109
Transkribus German Kurrent	German	guent...	PyLaia H...	11.09...	320
Transkribus German Kurre...	German	guent...	PyLaia H...	27.04...	320
DAT 18. Jh M3b_PyLaia	German	anne-c...	PyLaia H...	03.12...	102
German_Kurrent_17th-18th	german; lat...	alverm...	PyLaia H...	10.11...	183
Acta_17 PyLaia	german; lat...	alverm...	PyLaia H...	04.11...	594
German_Kurrent_XIX_pylaia	German	tobias...	PyLaia H...	11.09...	510

25 | Filter

Details

Name: Generic Model 15th-16th century German (prc) | Language: German

Description: The model is based on a selection of late medieval German manuscripts from the fifteenth and sixteenth centuries. | Parameters:

Document Type: Handwritten | Show advanced parameters...

Nr. of Words: 79204 | Nr. of Lines: 8240

Visibility: Public model with private data sets

Rating: - | ☐ Featured Model

Applications:

Save | Show Train Set | Show Validation Set | Show Characters

Learning Curve

CER

Epochs

— CER Train — CER Validation

CER on Train Set: 4.10% | CER on Validation Set: 5.60%

OK | Cancel

Figure 9 Models window

On the left side of the window, you can see an overview of the available models. On the top right, the details of the selected model are shown, and on the bottom right, the learning curve of your model. More information about these statistics can be found below.

Statistics

The “Learning Curve” graph signifies the accuracy of your model.

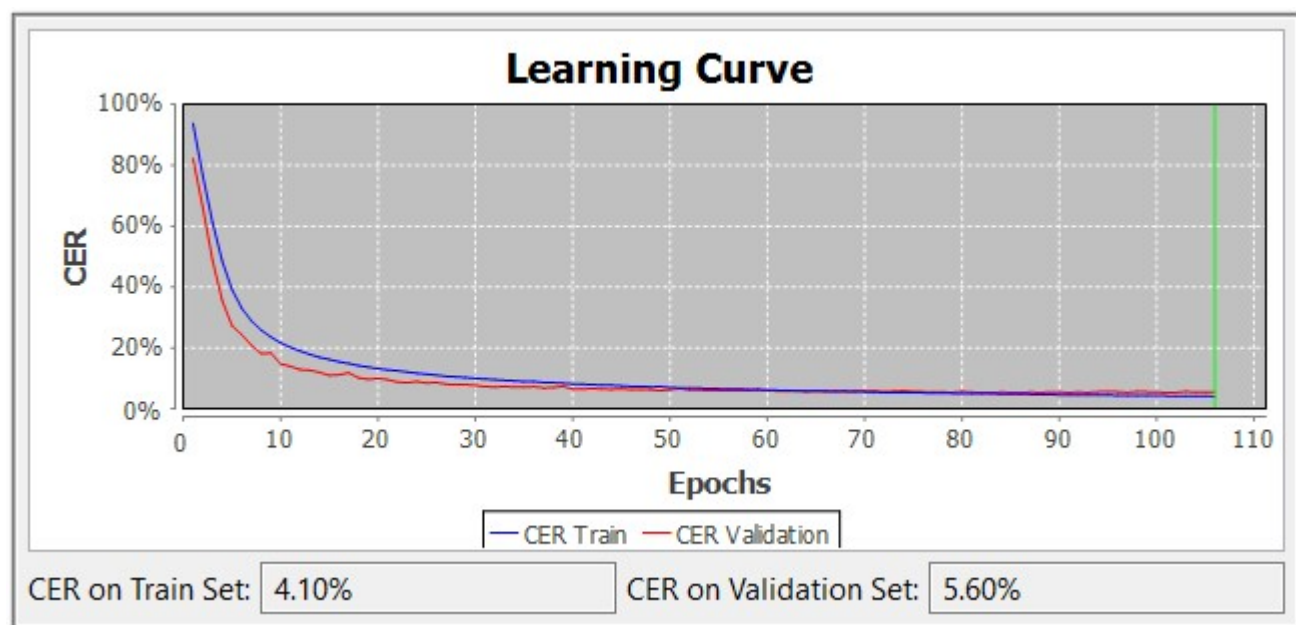




Figure 10. Learning Curve of your model

As you can see in Figure 10, the y-axis is defined as “Accuracy in CER”. “CER” stands for **Character Error Rate**, i.e. the percentage of characters that have been transcribed incorrectly by Text Recognition model.

“**Accuracy in CER**” is indicated as a percentage on the y-axis. The curve will always start at 100% and will go down as the training progresses, and the model improves.

The x-axis is defined as “**Epochs**”. During the training process, Transkribus will make an evaluation after every epoch. In Figure 10, the “Training Set” was divided into 106 epochs. When you train a model you can indicate how many “epochs” the “Training Set” should be divided into. The more epochs there are, the longer the training will take.

The **graph** shows two lines, one in blue and one in red. The **blue line** represents the progress of the training. The **red line** represents the progress of evaluations on the Validation Set. First the program trains itself on the **Training Set**, then it will test itself on pages in the **Validation Set**.

Underneath the graph, two percentage values are shown relating to the CER for the Training Set and the Validation Set. In Figure 10, the model performs with a 4.10% CER on the Training Set and 15.60% on the Validation Set.  

The value for the Validation Set is the most significant as it shows how the Text Model performs on pages that it has not been trained on. Results with a CER of 10% or below can be seen as very efficient for automated transcription.

Results with a CER of 20-30% are sufficient to work with powerful searching tools like Smart Search. For more details, see our How to Search Documents with Smart Search (<https://readcoop.eu/transkribus/howto/how-to-search-documents-with-smart-search/>).

Text Recognition

Now that you have your model, you can use it to automatically generate transcripts of the documents in your collection.

First, **upload** your documents to Transkribus. Second, **segment** your documents into text regions, lines and baselines. For more information on **uploading** and **segmentation**, please consult How To Transcribe Documents with Transkribus – Introduction. (https://transkribus.eu/wiki/index.php/How_to_Guides)

To access your model, click on the “Tools” tab and go to the “Text Recognition” section. Click “Run”, then click “Choose HTR-model”. Click on “Select HTR model” to choose your HTR model from the list of available models (your models plus the public ones) and click OK. Select whether you wish to generate a HTR transcript of one page or several pages. Press “Run” to start the text recognition process.

Once the recognition is finished, the automated transcription will appear in the text editor field.

Language Models

Language models are created automatically during the HTR-model training and can be added to the recognition process. The effect of language models needs to be tested in the individual case: in many cases, they are able to improve the recognition, but so far we also see cases, where they don't.

When selecting the HTR model, you can find the language model option top right. Click on the drop-down menu and choose “Language model from training data”.

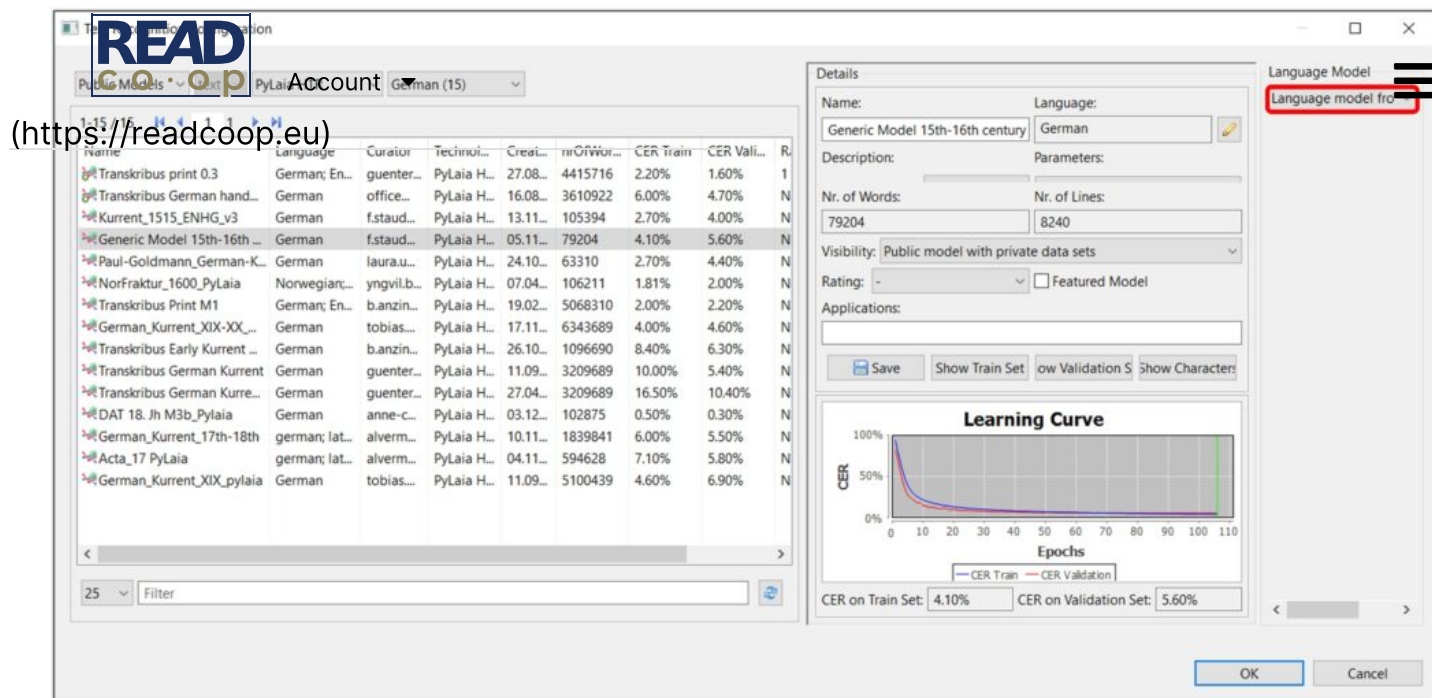


Figure 11. Language models

Share a model

You can share your HTR model with other collections in Transkribus, whether they are owned by you or by other users. If you want to share your model with another collection, you must have access to that collection (i.e. you have created the collection, or it has been shared with you).

In the “View Models” window, right-click on the name of your model. Then select “Share model...”

A window will open up, and by clicking on the green plus button, you can add the collection(s) to which you would like to share the model.

Your output

- As soon as the training is finished, you can try out your model on any other historical document with similar writing.
- You can share your model with other people who can benefit from it too.
- You can repeat the training process with more data in order to generate more efficient results.



You can measure the accuracy of your model with the “Compute Accuracy” function (<https://readcoop.eu/transkribus/howto/how-to-compute-accuracy-of-htr-models/>). The results of the HTR will depend on how similar and how clear the writing in the historical document is.

Credits

We would like to thank the many users who have contributed their feedback to help improve the Transkribus software.

The COOP

About us (<https://readcoop.eu/about/>)

Join us! (<https://readcoop.eu/join/>)

Our Members (<https://readcoop.eu/members/>)

Success Stories (<https://readcoop.eu/success-stories/>)

Work with us (<https://readcoop.eu/work-with-us/>)

Products & Services

Transkribus (<https://readcoop.eu/transkribus/>)

Transkribus lite (<https://transkribus.eu/lite>)

Read&search (<https://readcoop.eu/readsearch/>)

ScanTent (<https://readcoop.eu/scantent/>)

Useful information

News (<https://readcoop.eu/news/>)



Account ▼

Download Transkribus (<https://readcoop.eu/transkribus/download/>)

(<https://readcoop.eu>)

Public Models (<https://readcoop.eu/transkribus/public-models/>)

Payment and shipping (<https://readcoop.eu/payment-and-shipping/>)



Helpful resources

Resource center (<https://readcoop.eu/transkribus/resources/>)

How-to Guides (<https://readcoop.eu/transkribus/resources/how-to-guides/>)

Getting started with Transkribus (<https://readcoop.eu/transkribus/start/>)

FAQs (<https://readcoop.eu/transkribus/questions/>)

Videos (<https://readcoop.eu/transkribus/resources/videos/>)



Copyright © 2023 READ-COOP SCE

[Terms & conditions \(https://readcoop.eu/terms-and-conditions/\)](https://readcoop.eu/terms-and-conditions/)

[Privacy Policy \(https://readcoop.eu/privacy-policy/\)](https://readcoop.eu/privacy-policy/)

[Contact \(https://readcoop.eu/contact/\)](https://readcoop.eu/contact/)

[Imprint \(https://readcoop.eu/imprint/\)](https://readcoop.eu/imprint/)

EN