


Advanced-Methods-of-AI (/github/prakunjpratapsingh/Advanced-Methods-of-AI/tree/main)

/


AI_System_to_Digitize_Handwritten_Exams_in_German_Language_Studies.ipynb (/github/prakunjpratapsingh/Advanced-Methods-of-AI/tree/main/AI_System_to_Digitize_Handwritten_Exams_in_

 [Open in Colab](https://colab.research.google.com/github/prakunjpratapsingh/Advanced-Methods-of-AI/blob/main/AI_System_to_Digitize_Handwritten_Exams_in_German_Language_Studies.ipynb) (https://colab.research.google.com/github/prakunjpratapsingh/Advanced-Methods-of-AI/blob/main/AI_System_to_Digitize_Handwritten_Exams_in_German_Language_Studies.ipynb)

```
In [ ]: !pip install datasets transformers evaluate jiwer
```

```
Collecting datasets
  Downloading datasets-3.3.2-py3-none-any.whl.metadata (19 kB)
Requirement already satisfied: transformers in /usr/local/lib/python3.11/dist-packages (4.48.3)
Collecting evaluate
  Downloading evaluate-0.4.3-py3-none-any.whl.metadata (9.2 kB)
Collecting jiwer
  Downloading jiwer-3.1.0-py3-none-any.whl.metadata (2.6 kB)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from datasets) (3.17.0)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist-packages (from datasets) (1.26.4)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (17.0.0)
Collecting dill<0.3.9,>=0.3.0 (from datasets)
  Downloading dill-0.3.8-py3-none-any.whl.metadata (10 kB)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (from datasets) (2.2.2)
Requirement already satisfied: requests>=2.32.2 in /usr/local/lib/python3.11/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.11/dist-packages (from datasets) (4.67.1)
Collecting xxhash (from datasets)
  Downloading xxhash-3.5.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (12 kB)
Collecting multiprocess<0.70.17 (from datasets)
  Downloading multiprocess-0.70.16-py311-none-any.whl.metadata (7.2 kB)
Requirement already satisfied: fsspec<=2024.12.0,>=2023.1.0 in /usr/local/lib/python3.11/dist-packages (from fsspec[http]<=2024.12.0,>=2023.1.0->datasets) (2024.10.0)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.11/dist-packages (from datasets) (3.11.12)
Requirement already satisfied: huggingface-hub>=0.24.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.28.1)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.11/dist-packages (from transformers) (2024.11.6)
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.21.0)
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.5.2)
Requirement already satisfied: click>=8.1.8 in /usr/local/lib/python3.11/dist-packages (from jiwer) (8.1.8)
Collecting rapidfuzz>=3.9.7 (from jiwer)
  Downloading rapidfuzz-3.12.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (11 kB)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (2.4.6)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (1.3.2)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (25.1.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (6.1.0)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (0.2.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (1.18.3)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.24.0->datasets) (4.12.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (2.3.0)
```

```
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.3
2.2->datasets) (2025.1.31)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas->d
atasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets)
(2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets)
(2025.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2-
>pandas->datasets) (1.17.0)
Downloading datasets-3.3.2-py3-none-any.whl (485 kB)
_____ 485.4/485.4 kB 16.6 MB/s eta 0:00:00
Downloading evaluate-0.4.3-py3-none-any.whl (84 kB)
_____ 84.0/84.0 kB 7.9 MB/s eta 0:00:00
Downloading jiwer-3.1.0-py3-none-any.whl (22 kB)
Downloading dill-0.3.8-py3-none-any.whl (116 kB)
_____ 116.3/116.3 kB 9.6 MB/s eta 0:00:00
Downloading multiprocessing-0.70.16-py311-none-any.whl (143 kB)
_____ 143.5/143.5 kB 12.1 MB/s eta 0:00:00
Downloading rapidfuzz-3.12.1-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (3.1 MB)
_____ 3.1/3.1 MB 37.4 MB/s eta 0:00:00
Downloading xxhash-3.5.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (194 kB)
_____ 194.8/194.8 kB 13.4 MB/s eta 0:00:00
Installing collected packages: xxhash, rapidfuzz, dill, multiprocessing, jiwer, datasets, evaluate
Successfully installed datasets-3.3.2 dill-0.3.8 evaluate-0.4.3 jiwer-3.1.0 multiprocessing-0.70.16 rapidfuzz-3.1
2.1 xxhash-3.5.0
```

```
In [1]: # 1) One-time install (~ 5 min / ~1.4 GB)
!apt-get update -qq
!apt-get install -y texlive-xetex texlive-fonts-recommended 
          texlive-plain-generic pandoc # XeLaTeX + fonts + Pandoc
```

```
W: Skipping acquire of configured file 'main/source/Sources' as repository 'https://r2u.stat.illinois.edu/ubuntu
jammy InRelease' does not seem to provide it (sources.list entry misspelt?)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  dvisvgm fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono
  fonts-texgyre fonts-urw-base35 libapache-pom-java
  libcmark-gfm-extensions0.29.0.gfm.3 libcmark-gfm0.29.0.gfm.3
  libcommons-logging-java libcommons-parent-java libfontbox-java libgs9
  libgs9-common libidn12 libijs-0.35 libjbig2dec0 libkpathsea6 libpdfbox-java
  libptexenc1 libruby3.0 libsyntax2 libteckit0 libtexlua53 libtexluajit2
  libwoff1 libzzip-0-13 lmodern pandoc-data poppler-data preview-latex-style
  rake ruby ruby-net-telnet ruby-rubygems ruby-webrick ruby-xmlrpc ruby3.0
  rubygems-integration tlutils teckit tex-common tex-gyre texlive-base
  texlive-binaries texlive-latex-base texlive-latex-extra
  texlive-latex-recommended texlive-pictures tipa xfonts-encodings
  xfonts-utils
Suggested packages:
  fonts-noto fonts-freefont-otf | fonts-freefont-ttf libavalon-framework-java
  libcommons-logging-java-doc libexcalibur-logkit-java liblog4j1.2-java
  texlive-luatex pandoc-citeproc context wkhtmltopdf librsvg2-bin groff ghc
  nodejs php python libjs-mathjax libjs-katex citation-style-language-styles
  poppler-utils ghostscript fonts-japanese-mincho | fonts-ipafont-mincho
  fonts-japanese-gothic | fonts-ipafont-gothic fonts-arphic-ukai
  fonts-arphic-uming fonts-nanum ri ruby-dev bundler debhelper gv
  | postscript-viewer perl-tk xpdf | pdf-viewer xzdec
  texlive-fonts-recommended-doc texlive-latex-base-doc python3-pygments
  icc-profiles libfile-which-perl libspreadsheet-parseexcel-perl
  texlive-latex-extra-doc texlive-latex-recommended-doc texlive-pstricks
  dot2tex prerex texlive-pictures-doc vprerex default-jre-headless tipa-doc
The following NEW packages will be installed:
  dvisvgm fonts-droid-fallback fonts-lato fonts-lmodern fonts-noto-mono
  fonts-texgyre fonts-urw-base35 libapache-pom-java
  libcmark-gfm-extensions0.29.0.gfm.3 libcmark-gfm0.29.0.gfm.3
  libcommons-logging-java libcommons-parent-java libfontbox-java libgs9
  libgs9-common libidn12 libijs-0.35 libjbig2dec0 libkpathsea6 libpdfbox-java
  libptexenc1 libruby3.0 libsyntax2 libteckit0 libtexlua53 libtexluajit2
  libwoff1 libzzip-0-13 lmodern pandoc pandoc-data poppler-data
  preview-latex-style rake ruby ruby-net-telnet ruby-rubygems ruby-webrick
  ruby-xmlrpc ruby3.0 rubygems-integration tlutils teckit tex-common tex-gyre
  texlive-base texlive-binaries texlive-fonts-recommended texlive-latex-base
  texlive-latex-extra texlive-latex-recommended texlive-pictures
  texlive-plain-generic texlive-xetex tipa xfonts-encodings xfonts-utils
0 upgraded, 57 newly installed, 0 to remove and 51 not upgraded.
Need to get 202 MB of archives.
After this operation, 728 MB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-droid-fallback all 1:6.0.1r16-1.1build1 [1,805 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-lato all 2.0-2.1 [2,696 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy/main amd64 poppler-data all 0.4.11-1 [2,171 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy/universe amd64 tex-common all 6.17 [33.7 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-urw-base35 all 20200910-1 [6,367 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libgs9-common all 9.55.0~dfsg1-0ubuntu5.12 [753
kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libidn12 amd64 1.38-4ubuntu1 [60.0 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy/main amd64 libijs-0.35 amd64 0.35-15build2 [16.5 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy/main amd64 libjbig2dec0 amd64 0.19-3build2 [64.7 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libgs9 amd64 9.55.0~dfsg1-0ubuntu5.12 [5,031 k
B]
```

```
Get:11 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libkpathsea6 amd64 2021.20210626.59705-1ubuntu
0.2 [60.4 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/main amd64 libwoff1 amd64 1.0.2-1build4 [45.2 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy/universe amd64 dvisvgm amd64 2.13.1-1 [1,221 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy/universe amd64 fonts-lmodern all 2.004.5-6.1 [4,532 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-noto-mono all 20201225-1build1 [397 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy/universe amd64 fonts-texgyre all 20180621-3.1 [10.2 MB]
Get:17 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libapache-pom-java all 18-1 [4,720 B]
Get:18 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libcmr-gfm0.29.0.gfm.3 amd64 0.29.0.gfm.3-3 [115
kB]
Get:19 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libcmr-gfm-extensions0.29.0.gfm.3 amd64 0.29.0.gf
m.3-3 [25.1 kB]
Get:20 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libcommons-parent-java all 43-1 [10.8 kB]
Get:21 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libcommons-logging-java all 1.2-2 [60.3 kB]
Get:22 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libptexenc1 amd64 2021.20210626.59705-1ubuntu
0.2 [39.1 kB]
Get:23 http://archive.ubuntu.com/ubuntu jammy/main amd64 rubygems-integration all 1.18 [5,336 B]
Get:24 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 ruby3.0 amd64 3.0.2-7ubuntu2.10 [50.1 kB]
Get:25 http://archive.ubuntu.com/ubuntu jammy/main amd64 ruby-rubygems all 3.3.5-2 [228 kB]
Get:26 http://archive.ubuntu.com/ubuntu jammy/main amd64 ruby amd64 1:3.0~exp1 [5,100 B]
Get:27 http://archive.ubuntu.com/ubuntu jammy/main amd64 rake all 13.0.6-2 [61.7 kB]
Get:28 http://archive.ubuntu.com/ubuntu jammy/main amd64 ruby-net-telnet all 0.1.1-2 [12.6 kB]
Get:29 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 ruby-webrick all 1.7.0-3ubuntu0.1 [52.1 kB]
Get:30 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 ruby-xmllrpc all 0.3.2-1ubuntu0.1 [24.9 kB]
Get:31 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libruby3.0 amd64 3.0.2-7ubuntu2.10 [5,114 kB]
Get:32 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libsynctex2 amd64 2021.20210626.59705-1ubuntu
0.2 [55.6 kB]
Get:33 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libteckit0 amd64 2.5.11+ds1-1 [421 kB]
Get:34 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libtexlua53 amd64 2021.20210626.59705-1ubuntu
0.2 [120 kB]
Get:35 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libtexluaajit2 amd64 2021.20210626.59705-1ubuntu
0.2 [267 kB]
Get:36 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libzzip-0-13 amd64 0.13.72+dfsg.1-1.1 [27.0 kB]
Get:37 http://archive.ubuntu.com/ubuntu jammy/main amd64 xfonts-encodings all 1:1.0.5-0ubuntu2 [578 kB]
Get:38 http://archive.ubuntu.com/ubuntu jammy/main amd64 xfonts-utils amd64 1:7.7+6build2 [94.6 kB]
Get:39 http://archive.ubuntu.com/ubuntu jammy/universe amd64 lmodern all 2.004.5-6.1 [9,471 kB]
Get:40 http://archive.ubuntu.com/ubuntu jammy/universe amd64 pandoc-data all 2.9.2.1-3ubuntu2 [81.8 kB]
Get:41 http://archive.ubuntu.com/ubuntu jammy/universe amd64 pandoc amd64 2.9.2.1-3ubuntu2 [20.3 MB]
Get:42 http://archive.ubuntu.com/ubuntu jammy/universe amd64 preview-latex-style all 12.2-1ubuntu1 [185 kB]
Get:43 http://archive.ubuntu.com/ubuntu jammy/main amd64 tlutils amd64 1.41-4build2 [61.3 kB]
Get:44 http://archive.ubuntu.com/ubuntu jammy/universe amd64 teckit amd64 2.5.11+ds1-1 [699 kB]
Get:45 http://archive.ubuntu.com/ubuntu jammy/universe amd64 tex-gyre all 20180621-3.1 [6,209 kB]
Get:46 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 texlive-binaries amd64 2021.20210626.5970
5-1ubuntu0.2 [9,860 kB]
Get:47 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-base all 2021.20220204-1 [21.0 MB]
Get:48 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-fonts-recommended all 2021.20220204-1 [4,97
2 kB]
Get:49 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-latex-base all 2021.20220204-1 [1,128 kB]
Get:50 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libfontbox-java all 1:1.8.16-2 [207 kB]
Get:51 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libpdfbox-java all 1:1.8.16-2 [5,199 kB]
Get:52 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-latex-recommended all 2021.20220204-1 [14.4
MB]
Get:53 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-pictures all 2021.20220204-1 [8,720 kB]
Get:54 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-latex-extra all 2021.20220204-1 [13.9 MB]
Get:55 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-plain-generic all 2021.20220204-1 [27.5 MB]
Get:56 http://archive.ubuntu.com/ubuntu jammy/universe amd64 tipa all 2:1.3-21 [2,967 kB]
Get:57 http://archive.ubuntu.com/ubuntu jammy/universe amd64 texlive-xetex all 2021.20220204-1 [12.4 MB]
Fetched 202 MB in 10s (21.1 MB/s)
Extracting templates from packages: 100%
```

```
Preconfiguring packages ...
Selecting previously unselected package fonts-droid-fallback.
(Reading database ... 126281 files and directories currently installed.)
Preparing to unpack .../00-fonts-droid-fallback_1%3a6.0.1r16-1.1build1_all.deb ...
Unpacking fonts-droid-fallback (1:6.0.1r16-1.1build1) ...
Selecting previously unselected package fonts-lato.
Preparing to unpack .../01-fonts-lato_2.0-2.1_all.deb ...
Unpacking fonts-lato (2.0-2.1) ...
Selecting previously unselected package poppler-data.
Preparing to unpack .../02-poppler-data_0.4.11-1_all.deb ...
Unpacking poppler-data (0.4.11-1) ...
Selecting previously unselected package tex-common.
Preparing to unpack .../03-tex-common_6.17_all.deb ...
Unpacking tex-common (6.17) ...
Selecting previously unselected package fonts-urw-base35.
Preparing to unpack .../04-fonts-urw-base35_20200910-1_all.deb ...
Unpacking fonts-urw-base35 (20200910-1) ...
Selecting previously unselected package libgs9-common.
Preparing to unpack .../05-libgs9-common_9.55.0~dfsg1-0ubuntu5.12_all.deb ...
Unpacking libgs9-common (9.55.0~dfsg1-0ubuntu5.12) ...
Selecting previously unselected package libidn12:amd64.
Preparing to unpack .../06-libidn12_1.38-4ubuntu1_amd64.deb ...
Unpacking libidn12:amd64 (1.38-4ubuntu1) ...
Selecting previously unselected package libijs-0.35:amd64.
Preparing to unpack .../07-libijs-0.35_0.35-15build2_amd64.deb ...
Unpacking libijs-0.35:amd64 (0.35-15build2) ...
Selecting previously unselected package libjbig2dec0:amd64.
Preparing to unpack .../08-libjbig2dec0_0.19-3build2_amd64.deb ...
Unpacking libjbig2dec0:amd64 (0.19-3build2) ...
Selecting previously unselected package libgs9:amd64.
Preparing to unpack .../09-libgs9_9.55.0~dfsg1-0ubuntu5.12_amd64.deb ...
Unpacking libgs9:amd64 (9.55.0~dfsg1-0ubuntu5.12) ...
Selecting previously unselected package libkpathsea6:amd64.
Preparing to unpack .../10-libkpathsea6_2021.20210626.59705-1ubuntu0.2_amd64.deb ...
Unpacking libkpathsea6:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Selecting previously unselected package libwoff1:amd64.
Preparing to unpack .../11-libwoff1_1.0.2-1build4_amd64.deb ...
Unpacking libwoff1:amd64 (1.0.2-1build4) ...
Selecting previously unselected package dvisvgm.
Preparing to unpack .../12-dvisvgm_2.13.1-1_amd64.deb ...
Unpacking dvisvgm (2.13.1-1) ...
Selecting previously unselected package fonts-lmodern.
Preparing to unpack .../13-fonts-lmodern_2.004.5-6.1_all.deb ...
Unpacking fonts-lmodern (2.004.5-6.1) ...
Selecting previously unselected package fonts-noto-mono.
Preparing to unpack .../14-fonts-noto-mono_20201225-1build1_all.deb ...
Unpacking fonts-noto-mono (20201225-1build1) ...
Selecting previously unselected package fonts-texgyre.
Preparing to unpack .../15-fonts-texgyre_20180621-3.1_all.deb ...
Unpacking fonts-texgyre (20180621-3.1) ...
Selecting previously unselected package libapache-pom-java.
Preparing to unpack .../16-libapache-pom-java_18-1_all.deb ...
Unpacking libapache-pom-java (18-1) ...
Selecting previously unselected package libcmark-gfm0.29.0.gfm.3:amd64.
Preparing to unpack .../17-libcmark-gfm0.29.0.gfm.3_0.29.0.gfm.3-3_amd64.deb ...
Unpacking libcmark-gfm0.29.0.gfm.3:amd64 (0.29.0.gfm.3-3) ...
Selecting previously unselected package libcmark-gfm-extensions0.29.0.gfm.3:amd64.
Preparing to unpack .../18-libcmark-gfm-extensions0.29.0.gfm.3_0.29.0.gfm.3-3_amd64.deb ...
Unpacking libcmark-gfm-extensions0.29.0.gfm.3:amd64 (0.29.0.gfm.3-3) ...
```

```
Selecting previously unselected package libcommons-parent-java.
Preparing to unpack .../19-libcommons-parent-java_43-1_all.deb ...
Unpacking libcommons-parent-java (43-1) ...
Selecting previously unselected package libcommons-logging-java.
Preparing to unpack .../20-libcommons-logging-java_1.2-2_all.deb ...
Unpacking libcommons-logging-java (1.2-2) ...
Selecting previously unselected package libptexenc1:amd64.
Preparing to unpack .../21-libptexenc1_2021.20210626.59705-1ubuntu0.2_amd64.deb ...
Unpacking libptexenc1:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Selecting previously unselected package rubygems-integration.
Preparing to unpack .../22-rubygems-integration_1.18_all.deb ...
Unpacking rubygems-integration (1.18) ...
Selecting previously unselected package ruby3.0.
Preparing to unpack .../23-ruby3.0_3.0.2-7ubuntu2.10_amd64.deb ...
Unpacking ruby3.0 (3.0.2-7ubuntu2.10) ...
Selecting previously unselected package ruby-rubygems.
Preparing to unpack .../24-ruby-rubygems_3.3.5-2_all.deb ...
Unpacking ruby-rubygems (3.3.5-2) ...
Selecting previously unselected package ruby.
Preparing to unpack .../25-ruby_1%3a3.0~exp1_amd64.deb ...
Unpacking ruby (1:3.0~exp1) ...
Selecting previously unselected package rake.
Preparing to unpack .../26-rake_13.0.6-2_all.deb ...
Unpacking rake (13.0.6-2) ...
Selecting previously unselected package ruby-net-telnet.
Preparing to unpack .../27-ruby-net-telnet_0.1.1-2_all.deb ...
Unpacking ruby-net-telnet (0.1.1-2) ...
Selecting previously unselected package ruby-webrick.
Preparing to unpack .../28-ruby-webrick_1.7.0-3ubuntu0.1_all.deb ...
Unpacking ruby-webrick (1.7.0-3ubuntu0.1) ...
Selecting previously unselected package ruby-xmlrpc.
Preparing to unpack .../29-ruby-xmlrpc_0.3.2-1ubuntu0.1_all.deb ...
Unpacking ruby-xmlrpc (0.3.2-1ubuntu0.1) ...
Selecting previously unselected package libruby3.0:amd64.
Preparing to unpack .../30-libruby3.0_3.0.2-7ubuntu2.10_amd64.deb ...
Unpacking libruby3.0:amd64 (3.0.2-7ubuntu2.10) ...
Selecting previously unselected package libsyntax2:amd64.
Preparing to unpack .../31-libsyntax2_2021.20210626.59705-1ubuntu0.2_amd64.deb ...
Unpacking libsyntax2:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Selecting previously unselected package libteckit0:amd64.
Preparing to unpack .../32-libteckit0_2.5.11+ds1-1_amd64.deb ...
Unpacking libteckit0:amd64 (2.5.11+ds1-1) ...
Selecting previously unselected package libtexlua53:amd64.
Preparing to unpack .../33-libtexlua53_2021.20210626.59705-1ubuntu0.2_amd64.deb ...
Unpacking libtexlua53:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Selecting previously unselected package libtexluajit2:amd64.
Preparing to unpack .../34-libtexluajit2_2021.20210626.59705-1ubuntu0.2_amd64.deb ...
Unpacking libtexluajit2:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Selecting previously unselected package libzip-0-13:amd64.
Preparing to unpack .../35-libzip-0-13_0.13.72+dfsg.1-1.1_amd64.deb ...
Unpacking libzip-0-13:amd64 (0.13.72+dfsg.1-1.1) ...
Selecting previously unselected package xfonts-encodings.
Preparing to unpack .../36-xfonts-encodings_1%3a1.0.5-0ubuntu2_all.deb ...
Unpacking xfonts-encodings (1:1.0.5-0ubuntu2) ...
Selecting previously unselected package xfonts-utils.
Preparing to unpack .../37-xfonts-utils_1%3a7.7+6build2_amd64.deb ...
Unpacking xfonts-utils (1:7.7+6build2) ...
Selecting previously unselected package lmodern.
Preparing to unpack .../38-lmodern_2.004.5-6.1_all.deb ...
```

```
Unpacking lmodern (2.004.5-6.1) ...
Selecting previously unselected package pandoc-data.
Preparing to unpack .../39-pandoc-data_2.9.2.1-3ubuntu2_all.deb ...
Unpacking pandoc-data (2.9.2.1-3ubuntu2) ...
Selecting previously unselected package pandoc.
Preparing to unpack .../40-pandoc_2.9.2.1-3ubuntu2_amd64.deb ...
Unpacking pandoc (2.9.2.1-3ubuntu2) ...
Selecting previously unselected package preview-latex-style.
Preparing to unpack .../41-preview-latex-style_12.2-1ubuntu1_all.deb ...
Unpacking preview-latex-style (12.2-1ubuntu1) ...
Selecting previously unselected package tlutils.
Preparing to unpack .../42-tlutils_1.41-4build2_amd64.deb ...
Unpacking tlutils (1.41-4build2) ...
Selecting previously unselected package teckit.
Preparing to unpack .../43-teckit_2.5.11+ds1-1_amd64.deb ...
Unpacking teckit (2.5.11+ds1-1) ...
Selecting previously unselected package tex-gyre.
Preparing to unpack .../44-tex-gyre_20180621-3.1_all.deb ...
Unpacking tex-gyre (20180621-3.1) ...
Selecting previously unselected package texlive-binaries.
Preparing to unpack .../45-texlive-binaries_2021.20210626.59705-1ubuntu0.2_amd64.deb ...
Unpacking texlive-binaries (2021.20210626.59705-1ubuntu0.2) ...
Selecting previously unselected package texlive-base.
Preparing to unpack .../46-texlive-base_2021.20220204-1_all.deb ...
Unpacking texlive-base (2021.20220204-1) ...
Selecting previously unselected package texlive-fonts-recommended.
Preparing to unpack .../47-texlive-fonts-recommended_2021.20220204-1_all.deb ...
Unpacking texlive-fonts-recommended (2021.20220204-1) ...
Selecting previously unselected package texlive-latex-base.
Preparing to unpack .../48-texlive-latex-base_2021.20220204-1_all.deb ...
Unpacking texlive-latex-base (2021.20220204-1) ...
Selecting previously unselected package libfontbox-java.
Preparing to unpack .../49-libfontbox-java_1%3a1.8.16-2_all.deb ...
Unpacking libfontbox-java (1:1.8.16-2) ...
Selecting previously unselected package libpdfbox-java.
Preparing to unpack .../50-libpdfbox-java_1%3a1.8.16-2_all.deb ...
Unpacking libpdfbox-java (1:1.8.16-2) ...
Selecting previously unselected package texlive-latex-recommended.
Preparing to unpack .../51-texlive-latex-recommended_2021.20220204-1_all.deb ...
Unpacking texlive-latex-recommended (2021.20220204-1) ...
Selecting previously unselected package texlive-pictures.
Preparing to unpack .../52-texlive-pictures_2021.20220204-1_all.deb ...
Unpacking texlive-pictures (2021.20220204-1) ...
Selecting previously unselected package texlive-latex-extra.
Preparing to unpack .../53-texlive-latex-extra_2021.20220204-1_all.deb ...
Unpacking texlive-latex-extra (2021.20220204-1) ...
Selecting previously unselected package texlive-plain-generic.
Preparing to unpack .../54-texlive-plain-generic_2021.20220204-1_all.deb ...
Unpacking texlive-plain-generic (2021.20220204-1) ...
Selecting previously unselected package tipa.
Preparing to unpack .../55-tipa_2%3a1.3-21_all.deb ...
Unpacking tipa (2:1.3-21) ...
Selecting previously unselected package texlive-xetex.
Preparing to unpack .../56-texlive-xetex_2021.20220204-1_all.deb ...
Unpacking texlive-xetex (2021.20220204-1) ...
Setting up fonts-lato (2.0-2.1) ...
Setting up fonts-noto-mono (20201225-1build1) ...
Setting up libwoff1:amd64 (1.0.2-1build4) ...
Setting up libtexlua53:amd64 (2021.20210626.59705-1ubuntu0.2) ...
```



```
Setting up libijs-0.35:amd64 (0.35-15build2) ...
Setting up libtexlua32:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Setting up libfontbox-java (1:1.8.16-2) ...
Setting up rubygems-integration (1.18) ...
Setting up libzip-0-13:amd64 (0.13.72+dfsg.1-1.1) ...
Setting up fonts-urw-base35 (20200910-1) ...
Setting up poppler-data (0.4.11-1) ...
Setting up tex-common (6.17) ...
update-language: texlive-base not installed and configured, doing nothing!
Setting up libjbig2dec0:amd64 (0.19-3build2) ...
Setting up libteckit0:amd64 (2.5.11+ds1-1) ...
Setting up libapache-pom-java (18-1) ...
Setting up ruby-net-telnet (0.1.1-2) ...
Setting up xfonts-encodings (1:1.0.5-0ubuntu2) ...
Setting up tlutils (1.41-4build2) ...
Setting up libidn12:amd64 (1.38-4ubuntu1) ...
Setting up fonts-texgyre (20180621-3.1) ...
Setting up libkpathsea6:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Setting up ruby-webrick (1.7.0-3ubuntu0.1) ...
Setting up libcmark-gfm0.29.0.gfm.3:amd64 (0.29.0.gfm.3-3) ...
Setting up fonts-lmodern (2.004.5-6.1) ...
Setting up libcmark-gfm-extensions0.29.0.gfm.3:amd64 (0.29.0.gfm.3-3) ...
Setting up fonts-droid-fallback (1:6.0.1r16-1.1build1) ...
Setting up pandoc-data (2.9.2.1-3ubuntu2) ...
Setting up ruby-xmllrpc (0.3.2-1ubuntu0.1) ...
Setting up libsynctex2:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Setting up libgs9-common (9.55.0~dfsg1-0ubuntu5.12) ...
Setting up teckit (2.5.11+ds1-1) ...
Setting up libpdfbox-java (1:1.8.16-2) ...
Setting up libgs9:amd64 (9.55.0~dfsg1-0ubuntu5.12) ...
Setting up preview-latex-style (12.2-1ubuntu1) ...
Setting up libcommons-parent-java (43-1) ...
Setting up dvisvgm (2.13.1-1) ...
Setting up libcommons-logging-java (1.2-2) ...
Setting up xfonts-utils (1:7.7+6build2) ...
Setting up libptexenc1:amd64 (2021.20210626.59705-1ubuntu0.2) ...
Setting up pandoc (2.9.2.1-3ubuntu2) ...
Setting up texlive-binaries (2021.20210626.59705-1ubuntu0.2) ...
update-alternatives: using /usr/bin/xdvi-xaw to provide /usr/bin/xdvi.bin (xdvi.bin) in auto mode
update-alternatives: using /usr/bin/bibtex.original to provide /usr/bin/bibtex (bibtex) in auto mode
Setting up lmodern (2.004.5-6.1) ...
Setting up texlive-base (2021.20220204-1) ...
/usr/bin/ucfr
/usr/bin/ucfr
/usr/bin/ucfr
/usr/bin/ucfr
mktexlsr: Updating /var/lib/texmf/ls-R-TEXLIVEDIST...
mktexlsr: Updating /var/lib/texmf/ls-R-TEXMFMAIN...
mktexlsr: Updating /var/lib/texmf/ls-R...
mktexlsr: Done.
tl-paper: setting paper size for dvips to a4: /var/lib/texmf/dvips/config/config-paper.ps
tl-paper: setting paper size for dvipdfmx to a4: /var/lib/texmf/dvipdfmx/dvipdfmx-paper.cfg
tl-paper: setting paper size for xdvi to a4: /var/lib/texmf/xdvi/XDvi-paper
tl-paper: setting paper size for pdftex to a4: /var/lib/texmf/tex/generic/tex-ini-files/pdftexconfig.tex
Setting up tex-gyre (20180621-3.1) ...
Setting up texlive-plain-generic (2021.20220204-1) ...
Setting up texlive-latex-base (2021.20220204-1) ...
Setting up texlive-latex-recommended (2021.20220204-1) ...
Setting up texlive-pictures (2021.20220204-1) ...
```

```
Setting up texlive-fonts-recommended (2021.20220204-1) ...
Setting up tipa (2:1.3-21) ...
Setting up texlive-latex-extra (2021.20220204-1) ...
Setting up texlive-xetex (2021.20220204-1) ...
Setting up rake (13.0.6-2) ...
Setting up libruby3.0:amd64 (3.0.2-7ubuntu2.10) ...
Setting up ruby3.0 (3.0.2-7ubuntu2.10) ...
Setting up ruby (1:3.0~exp1) ...
Setting up ruby-rubygems (3.3.5-2) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for mailcap (3.70+nmlubuntu1) ...
Processing triggers for fontconfig (2.13.1-4.2ubuntu5) ...
Processing triggers for libc-bin (2.35-0ubuntu3.8) ...
/sbin/ldconfig.real: /usr/local/lib/libur_adapter_level_zero_v2.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libur_loader.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libumf.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbb.so.12 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libur_adapter_level_zero.so.0 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc_proxy.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_5.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtcm_debug.so.1 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_0.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libhwloc.so.15 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtcm.so.1 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libur_adapter_opencl.so.0 is not a symbolic link

Processing triggers for tex-common (6.17) ...
Running updmmap-sys. This may take some time... done.
Running mktexlsr /var/lib/texmf ... done.
Building format(s) --all.
    This may take some time... done.
```

```
In [2]: # 2) Convert any notebook already present in /content
!jupyter nbconvert --to pdf --output AI_System_to_Digitize_Handwritten_Exams_in_German_Language_Studies.pdf /con
```

```
[NbConvertApp] WARNING | pattern '/content/my_notebook.ipynb' matched no files
This application is used to convert notebook files (*.ipynb)
to various other formats.
```

```
WARNING: THE COMMANDLINE INTERFACE MAY CHANGE IN FUTURE RELEASES.
```

Options

=====

The options below are convenience aliases to configurable class-options,
as listed in the "Equivalent to" description-line of the aliases.

To see all configurable class-options for some <cmd>, use:

```
<cmd> --help-all
```

--debug

set log level to logging.DEBUG (maximize logging output)

Equivalent to: [--Application.log_level=10]

--show-config

Show the application's configuration (human-readable format)

Equivalent to: [--Application.show_config=True]

--show-config-json

Show the application's configuration (json format)

Equivalent to: [--Application.show_config_json=True]

--generate-config

generate default config file

Equivalent to: [--JupyterApp.generate_config=True]

-y

Answer yes to any questions instead of prompting.

Equivalent to: [--JupyterApp.answer_yes=True]

--execute

Execute the notebook prior to export.

Equivalent to: [--ExecutePreprocessor.enabled=True]

--allow-errors

Continue notebook execution even if one of the cells throws an error and include the error message in the cell output (the default behaviour is to abort conversion). This flag is only relevant if '--execute' was specified, too.

Equivalent to: [--ExecutePreprocessor.allow_errors=True]

--stdin

read a single notebook file from stdin. Write the resulting notebook with default basename 'notebook.*'

Equivalent to: [--NbConvertApp.from_stdin=True]

--stdout

Write notebook output to stdout instead of files.

Equivalent to: [--NbConvertApp.writer_class=StdoutWriter]

--inplace

Run nbconvert in place, overwriting the existing notebook (only

relevant when converting to notebook format)

Equivalent to: [--NbConvertApp.use_output_suffix=False --NbConvertApp.export_format=notebook --FilesWriter.build_directory=]

--clear-output

Clear output of current file and save in place,

overwriting the existing notebook.

Equivalent to: [--NbConvertApp.use_output_suffix=False --NbConvertApp.export_format=notebook --FilesWriter.build_directory= --ClearOutputPreprocessor.enabled=True]

--coalesce-streams

Coalesce consecutive stdout and stderr outputs into one stream (within each cell).

Equivalent to: [--NbConvertApp.use_output_suffix=False --NbConvertApp.export_format=notebook --FilesWriter.build_directory= --CoalesceStreamsPreprocessor.enabled=True]

--no-prompt

Exclude input and output prompts from converted document.

Equivalent to: [--TemplateExporter.exclude_input_prompt=True --TemplateExporter.exclude_output_prompt=True]

```

--no-input
  Exclude input cells and output prompts from converted document.
  This mode is ideal for generating code-free reports.
  Equivalent to: [--TemplateExporter.exclude_output_prompt=True --TemplateExporter.exclude_input=True --Template
teExporter.exclude_input_prompt=True]
--allow-chromium-download
  Whether to allow downloading chromium if no suitable version is found on the system.
  Equivalent to: [--WebPDFExporter.allow_chromium_download=True]
--disable-chromium-sandbox
  Disable chromium security sandbox when converting to PDF..
  Equivalent to: [--WebPDFExporter.disable_sandbox=True]
--show-input
  Shows code input. This flag is only useful for dejavu users.
  Equivalent to: [--TemplateExporter.exclude_input=False]
--embed-images
  Embed the images as base64 dataurls in the output. This flag is only useful for the HTML/WebPDF/Slides expor
ts.
  Equivalent to: [--HTMLExporter.embed_images=True]
--sanitize-html
  Whether the HTML in Markdown cells and cell outputs should be sanitized..
  Equivalent to: [--HTMLExporter.sanitize_html=True]
--log-level=<Enum>
  Set the log level by value or name.
  Choices: any of [0, 10, 20, 30, 40, 50, 'DEBUG', 'INFO', 'WARN', 'ERROR', 'CRITICAL']
  Default: 30
  Equivalent to: [--Application.log_level]
--config=<Unicode>
  Full path of a config file.
  Default: ''
  Equivalent to: [--JupyterApp.config_file]
--to=<Unicode>
  The export format to be used, either one of the built-in formats
  ['asciidoc', 'custom', 'html', 'latex', 'markdown', 'notebook', 'pdf', 'python', 'qtpdf', 'qtpng', '
rst', 'script', 'slides', 'webpdf']
  or a dotted object name that represents the import path for an
  ``Exporter`` class
  Default: ''
  Equivalent to: [--NbConvertApp.export_format]
--template=<Unicode>
  Name of the template to use
  Default: ''
  Equivalent to: [--TemplateExporter.template_name]
--template-file=<Unicode>
  Name of the template file to use
  Default: None
  Equivalent to: [--TemplateExporter.template_file]
--theme=<Unicode>
  Template specific theme(e.g. the name of a JupyterLab CSS theme distributed
as prebuilt extension for the lab template)
  Default: 'light'
  Equivalent to: [--HTMLExporter.theme]
--sanitize_html=<Bool>
  Whether the HTML in Markdown cells and cell outputs should be sanitized.This
should be set to True by nbviewer or similar tools.
  Default: False
  Equivalent to: [--HTMLExporter.sanitize_html]
--writer=<DottedObjectName>
  Writer class used to write the
                                results of the conversion

```

```

    Default: 'FilesWriter'
    Equivalent to: [--NbConvertApp.writer_class]
--post=<DottedOrNone>
    PostProcessor class used to write the
                                results of the conversion

    Default: ''
    Equivalent to: [--NbConvertApp.postprocessor_class]
--output=<Unicode>
    Overwrite base name use for output files.
    Supports pattern replacements '{notebook_name}'.

    Default: '{notebook_name}'
    Equivalent to: [--NbConvertApp.output_base]
--output-dir=<Unicode>
    Directory to write output(s) to. Defaults
                                to output to the directory of each notebook. To recover
                                previous default behaviour (outputting to the current
                                working directory) use . as the flag value.

    Default: ''
    Equivalent to: [--FilesWriter.build_directory]
--reveal-prefix=<Unicode>
    The URL prefix for reveal.js (version 3.x).
    This defaults to the reveal CDN, but can be any url pointing to a copy
    of reveal.js.
    For speaker notes to work, this must be a relative path to a local
    copy of reveal.js: e.g., "reveal.js".
    If a relative path is given, it must be a subdirectory of the
    current directory (from which the server is run).
    See the usage documentation
    (https://nbconvert.readthedocs.io/en/latest/usage.html#reveal-js-html-slideshow)
    for more details.

    Default: ''
    Equivalent to: [--SlidesExporter.reveal_url_prefix]
--nbformat=<Enum>
    The nbformat version to write.
    Use this to downgrade notebooks.
    Choices: any of [1, 2, 3, 4]
    Default: 4
    Equivalent to: [--NotebookExporter.nbformat_version]

```

Examples

The simplest way to use nbconvert is

```
> jupyter nbconvert mynotebook.ipynb --to html
```

Options include ['asciidoc', 'custom', 'html', 'latex', 'markdown', 'notebook', 'pdf', 'python', 'qt
pdf', 'qtpng', 'rst', 'script', 'slides', 'webpdf'].

```
> jupyter nbconvert --to latex mynotebook.ipynb
```

Both HTML and LaTeX support multiple output templates. LaTeX includes
'base', 'article' and 'report'. HTML includes 'basic', 'lab' and
'classic'. You can specify the flavor of the format used.

```
> jupyter nbconvert --to html --template lab mynotebook.ipynb
```

You can also pipe the output to stdout, rather than a file

```
> jupyter nbconvert mynotebook.ipynb --stdout
```

PDF is generated via latex

```
> jupyter nbconvert mynotebook.ipynb --to pdf
```

You can get (and serve) a Reveal.js-powered slideshow

```
> jupyter nbconvert myslides.ipynb --to slides --post serve
```

Multiple notebooks can be given at the command line in a couple of different ways:

```
> jupyter nbconvert notebook*.ipynb
> jupyter nbconvert notebook1.ipynb notebook2.ipynb
```

or you can specify the notebooks list in a config file, containing::

```
c.NbConvertApp.notebooks = ["my_notebook.ipynb"]
```

```
> jupyter nbconvert --config mycfg.py
```

To see all available configurables, use `--help-all`.

In []:

In []: ! pip install datasets

```
Requirement already satisfied: datasets in /usr/local/lib/python3.11/dist-packages (3.3.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from datasets) (3.17.0)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist-packages (from datasets) (1.26.4)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (17.0.0)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.3.8)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (from datasets) (2.2.2)
Requirement already satisfied: requests>=2.32.2 in /usr/local/lib/python3.11/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.11/dist-packages (from datasets) (4.67.1)
Requirement already satisfied: xxhash in /usr/local/lib/python3.11/dist-packages (from datasets) (3.5.0)
Requirement already satisfied: multiprocessing<0.70.17 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.70.16)
Requirement already satisfied: fsspec<=2024.12.0,>=2023.1.0 in /usr/local/lib/python3.11/dist-packages (from fsspec[http]<=2024.12.0,>=2023.1.0->datasets) (2024.10.0)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.11/dist-packages (from datasets) (3.11.12)
Requirement already satisfied: huggingface-hub>=0.24.0 in /usr/local/lib/python3.11/dist-packages (from datasets) (0.28.1)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (2.4.6)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (1.3.2)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (25.1.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (6.1.0)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (0.2.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets) (1.18.3)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.24.0->datasets) (4.12.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.32.2->datasets) (2025.1.31)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas->datasets) (2025.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas->datasets) (1.17.0)
```

```
In [ ]: !pip install evaluate
```

```
Requirement already satisfied: evaluate in /usr/local/lib/python3.11/dist-packages (0.4.3)
Requirement already satisfied: datasets>=2.0.0 in /usr/local/lib/python3.11/dist-packages (from evaluate) (3.3.2)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist-packages (from evaluate) (1.26.4)
Requirement already satisfied: dill in /usr/local/lib/python3.11/dist-packages (from evaluate) (0.3.8)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (from evaluate) (2.2.2)
Requirement already satisfied: requests>=2.19.0 in /usr/local/lib/python3.11/dist-packages (from evaluate) (2.32.3)
Requirement already satisfied: tqdm>=4.62.1 in /usr/local/lib/python3.11/dist-packages (from evaluate) (4.67.1)
Requirement already satisfied: xxhash in /usr/local/lib/python3.11/dist-packages (from evaluate) (3.5.0)
Requirement already satisfied: multiprocessing in /usr/local/lib/python3.11/dist-packages (from evaluate) (0.70.16)
Requirement already satisfied: fsspec>=2021.05.0 in /usr/local/lib/python3.11/dist-packages (from fsspec[http]>=2021.05.0->evaluate) (2024.10.0)
Requirement already satisfied: huggingface-hub>=0.7.0 in /usr/local/lib/python3.11/dist-packages (from evaluate) (0.28.1)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from evaluate) (24.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0->evaluate) (3.17.0)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0->evaluate) (17.0.0)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0->evaluate) (3.11.12)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0->evaluate) (6.0.2)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.7.0->evaluate) (4.12.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests>=2.19.0->evaluate) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.19.0->evaluate) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.19.0->evaluate) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.19.0->evaluate) (2025.1.31)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas->evaluate) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas->evaluate) (2025.1)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas->evaluate) (2025.1)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (2.4.6)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (1.3.2)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (25.1.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (6.1.0)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (0.2.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.11/dist-packages (from aiohttp->datasets>=2.0.0->evaluate) (1.18.3)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas->evaluate) (1.17.0)
```



```
In [ ]: import numpy as np
import pandas as pd
from datasets import load_dataset
from PIL import Image
from sklearn.model_selection import train_test_split
import torch
from torch.utils.data import Dataset
from transformers import TrOCRProcessor, VisionEncoderDecoderModel
import evaluate
from torch.optim import AdamW
from tqdm.notebook import tqdm
```

```
In [ ]: # Load CER metric
cer_metric = evaluate.load("cer")
```

```
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
Downloading builder script:   0%|          | 0.00/5.60k [00:00<?, ?B/s]
```

Dataset Description

```
In [ ]: # Load dataset
dataset = load_dataset('fhswf/german_handwriting')
df = pd.DataFrame(dataset['train'])

README.md:   0%|          | 0.00/893 [00:00<?, ?B/s]
train-00000-of-00008.parquet:   0%|          | 0.00/366M [00:00<?, ?B/s]
train-00001-of-00008.parquet:   0%|          | 0.00/18.8M [00:00<?, ?B/s]
train-00002-of-00008.parquet:   0%|          | 0.00/55.5M [00:00<?, ?B/s]
train-00003-of-00008.parquet:   0%|          | 0.00/67.7M [00:00<?, ?B/s]
train-00004-of-00008.parquet:   0%|          | 0.00/165M [00:00<?, ?B/s]
train-00005-of-00008.parquet:   0%|          | 0.00/559M [00:00<?, ?B/s]
train-00006-of-00008.parquet:   0%|          | 0.00/184M [00:00<?, ?B/s]
train-00007-of-00008.parquet:   0%|          | 0.00/381M [00:00<?, ?B/s]
Generating train split:   0%|          | 0/10854 [00:00<?, ? examples/s]
```

```
In [ ]: df.head()
```

```
Out[ ]:
   image text
0  <PIL.PngImagePlugin.PngImageFile image mode=RG...  - Terminvorschlag bis
1  <PIL.PngImagePlugin.PngImageFile image mode=RG...  - Module an Michael
2  <PIL.PngImagePlugin.PngImageFile image mode=RG...      Handschrift
3  <PIL.PngImagePlugin.PngImageFile image mode=RG...      Datenschutz bei
4  <PIL.PngImagePlugin.PngImageFile image mode=RG...      -> Urheberrecht
```

```
In [ ]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10854 entries, 0 to 10853
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype
---  -
 0   image   10854 non-null   object
 1   text    10844 non-null   object
dtypes: object(2)
memory usage: 169.7+ KB
```

```
In [ ]: df.isnull().sum()
```

```
Out[ ]:      0
image    0
text    10
```

dtype: int64

```
In [ ]: missing_text_rows = df[df['text'].isnull()]
print("Rows with missing 'text' values:")
print(missing_text_rows)
```

```
Rows with missing 'text' values:
      image  text
3060  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
4436  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
4536  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
6265  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
6540  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
7391  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
7944  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
9531  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
10206  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
10333  <PIL.PngImagePlugin.PngImageFile image mode=RG...  None
```

```
In [ ]: df = df.dropna(subset=['text'])
```

```
In [ ]: df.isnull().sum()
```

```
Out[ ]:      0
image    0
text     0
```

dtype: int64

```
In [ ]: class IAMDataset(Dataset):
    def __init__(self, df, processor, max_target_length=128):
        self.df = df
        self.processor = processor
        self.max_target_length = max_target_length

    def __len__(self):
        return len(self.df)

    def __getitem__(self, idx):
        image = self.df['image'][idx] # PIL.Image object
        text = self.df['text'][idx]

        # Process image
        processed_image = self.processor(image, return_tensors="pt")
        pixel_values = processed_image.pixel_values.squeeze(0)

        # Tokenize text
        labels = self.processor.tokenizer(
            text,
            padding="max_length",
            max_length=self.max_target_length,
            truncation=True,
        ).input_ids
        labels = [label if label != self.processor.tokenizer.pad_token_id else -100 for label in labels]

        return {"pixel_values": pixel_values, "labels": torch.tensor(labels)}
```

```
In [ ]: # Split dataset
train_df, test_df = train_test_split(df, test_size=0.2, random_state=42)
cleaned_train_df = train_df[train_df['text'].notnull() & (train_df['text'] != "")]
cleaned_train_df.reset_index(drop=True, inplace=True)
test_df.reset_index(drop=True, inplace=True)
```

```
In [ ]: # Validate and fix images
def validate_images(df):
    for idx, image in enumerate(df['image']):
        try:
            if not isinstance(image, Image.Image):
                raise ValueError(f"Image at index {idx} is not a valid PIL image.")
            if image.mode != "RGB":
                df['image'][idx] = image.convert("RGB")
        except Exception as e:
            print(f"Error at index {idx}: {e}")

validate_images(cleaned_train_df)
validate_images(test_df)
```

```
<ipython-input-15-ea72c002e636>:8: FutureWarning: ChainedAssignmentError: behaviour will change in pandas 3.0!
You are setting values through chained assignment. Currently this works in certain cases, but when using Copy-on-Write (which will become the default behaviour in pandas 3.0) this will never work to update the original Data
Frame or Series, because the intermediate object on which we are setting values will behave as a copy.
A typical example is when you are setting values in a column of a DataFrame, like:
```

```
df["col"][row_indexer] = value
```

```
Use `df.loc[row_indexer, "col"] = values` instead, to perform the assignment in a single step and ensure this keeps
updating the original `df`.
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

```
df['image'][idx] = image.convert("RGB")
```

```
In [ ]: # Instantiate the processor
processor = TrOCRProcessor.from_pretrained("microsoft/trocr-base-handwritten")

preprocessor_config.json:  0%|          | 0.00/224 [00:00<?, ?B/s]
Using a slow image processor as `use_fast` is unset and a slow processor was saved with this model. `use_fast=True` will be the default behavior in v4.48, even if the model was saved with a slow processor. This will result in minor differences in outputs. You'll still be able to use a slow processor with `use_fast=False`.
tokenizer_config.json:    0%|          | 0.00/1.12k [00:00<?, ?B/s]
vocab.json:              0%|          | 0.00/899k [00:00<?, ?B/s]
merges.txt:              0%|          | 0.00/456k [00:00<?, ?B/s]
special_tokens_map.json:  0%|          | 0.00/772 [00:00<?, ?B/s]
```

```
In [ ]: # Define datasets
train_dataset = IAMDataset(df=cleaned_train_df, processor=processor)
eval_dataset = IAMDataset(df=test_df, processor=processor)
```

```
In [ ]: # Collate function
def collate_fn(batch):
    pixel_values = torch.stack([item["pixel_values"] for item in batch])
    labels = torch.nn.utils.rnn.pad_sequence(
        [item["labels"] for item in batch], batch_first=True, padding_value=-100
    )
    return {"pixel_values": pixel_values, "labels": labels}
```

```
In [ ]: # Data loaders
train_dataloader = torch.utils.data.DataLoader(
    train_dataset, batch_size=8, shuffle=True, collate_fn=collate_fn
)
eval_dataloader = torch.utils.data.DataLoader(
    eval_dataset, batch_size=8, shuffle=False, collate_fn=collate_fn
)
```

Model Definition & Fine-tuning

```
In [ ]: # Initialize model
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
model = VisionEncoderDecoderModel.from_pretrained("microsoft/trocr-base-stage1")
model.to(device)

config.json:  0%|          | 0.00/4.21k [00:00<?, ?B/s]
model.safetensors: 0%|          | 0.00/1.54G [00:00<?, ?B/s]
```

```
Config of the encoder: <class 'transformers.models.vit.modeling_vit.ViTModel'> is overwritten by shared encoder
config: ViTConfig {
  "attention_probs_dropout_prob": 0.0,
  "encoder_stride": 16,
  "hidden_act": "gelu",
  "hidden_dropout_prob": 0.0,
  "hidden_size": 768,
  "image_size": 384,
  "initializer_range": 0.02,
  "intermediate_size": 3072,
  "layer_norm_eps": 1e-12,
  "model_type": "vit",
  "num_attention_heads": 12,
  "num_channels": 3,
  "num_hidden_layers": 12,
  "patch_size": 16,
  "qkv_bias": false,
  "transformers_version": "4.48.3"
}

Config of the decoder: <class 'transformers.models.trocr.modeling_trocr.TrOCRForCausalLM'> is overwritten by sha
red decoder config: TrOCRConfig {
  "activation_dropout": 0.0,
  "activation_function": "relu",
  "add_cross_attention": true,
  "attention_dropout": 0.0,
  "bos_token_id": 0,
  "classifier_dropout": 0.0,
  "cross_attention_hidden_size": 768,
  "d_model": 1024,
  "decoder_attention_heads": 16,
  "decoder_ffn_dim": 4096,
  "decoder_layerdrop": 0.0,
  "decoder_layers": 12,
  "decoder_start_token_id": 2,
  "dropout": 0.1,
  "eos_token_id": 2,
  "init_std": 0.02,
  "is_decoder": true,
  "layernorm_embedding": false,
  "max_position_embeddings": 1024,
  "model_type": "trocr",
  "pad_token_id": 1,
  "scale_embedding": true,
  "tie_word_embeddings": false,
  "transformers_version": "4.48.3",
  "use_cache": false,
  "use_learned_position_embeddings": false,
  "vocab_size": 50265
}

Some weights of VisionEncoderDecoderModel were not initialized from the model checkpoint at microsoft/trocr-bas
e-stagel and are newly initialized: ['encoder.pooler.dense.bias', 'encoder.pooler.dense.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
generation_config.json:  0%|          | 0.00/190 [00:00<?, ?B/s]
```

```

Out[ ]: VisionEncoderDecoderModel(
  (encoder): ViTModel(
    (embeddings): ViTEmbeddings(
      (patch_embeddings): ViTPatchEmbeddings(
        (projection): Conv2d(3, 768, kernel_size=(16, 16), stride=(16, 16))
      )
      (dropout): Dropout(p=0.0, inplace=False)
    )
    (encoder): ViTEncoder(
      (layer): ModuleList(
        (0-11): 12 x ViTLayer(
          (attention): ViTSdpaAttention(
            (attention): ViTSdpaSelfAttention(
              (query): Linear(in_features=768, out_features=768, bias=False)
              (key): Linear(in_features=768, out_features=768, bias=False)
              (value): Linear(in_features=768, out_features=768, bias=False)
              (dropout): Dropout(p=0.0, inplace=False)
            )
            (output): ViTSelfOutput(
              (dense): Linear(in_features=768, out_features=768, bias=True)
              (dropout): Dropout(p=0.0, inplace=False)
            )
          )
          (intermediate): ViTIntermediate(
            (dense): Linear(in_features=768, out_features=3072, bias=True)
            (intermediate_act_fn): GELUActivation()
          )
          (output): ViTOutput(
            (dense): Linear(in_features=3072, out_features=768, bias=True)
            (dropout): Dropout(p=0.0, inplace=False)
          )
          (layernorm_before): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
          (layernorm_after): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
        )
      )
    )
    (layernorm): LayerNorm((768,), eps=1e-12, elementwise_affine=True)
    (pooler): ViTPooler(
      (dense): Linear(in_features=768, out_features=768, bias=True)
      (activation): Tanh()
    )
  )
  (decoder): TrOCRForCausalLM(
    (model): TrOCRDecoderWrapper(
      (decoder): TrOCRDecoder(
        (embed_tokens): TrOCRScaledWordEmbedding(50265, 1024, padding_idx=1)
        (embed_positions): TrOCRSinusoidalPositionalEmbedding()
        (layers): ModuleList(
          (0-11): 12 x TrOCRDecoderLayer(
            (self_attn): TrOCRAttention(
              (k_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (v_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (q_proj): Linear(in_features=1024, out_features=1024, bias=True)
              (out_proj): Linear(in_features=1024, out_features=1024, bias=True)
            )
            (activation_fn): ReLU()
            (self_attn_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
            (encoder_attn): TrOCRAttention(
              (k_proj): Linear(in_features=768, out_features=1024, bias=True)

```

```

        (v_proj): Linear(in_features=768, out_features=1024, bias=True)
        (q_proj): Linear(in_features=1024, out_features=1024, bias=True)
        (out_proj): Linear(in_features=1024, out_features=1024, bias=True)
    )
    (encoder_attn_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
    (fc1): Linear(in_features=1024, out_features=4096, bias=True)
    (fc2): Linear(in_features=4096, out_features=1024, bias=True)
    (final_layer_norm): LayerNorm((1024,), eps=1e-05, elementwise_affine=True)
)
)
)
)
(output_projection): Linear(in_features=1024, out_features=50265, bias=False)
)
)

```

```

In [ ]: # Configure model
model.config.decoder_start_token_id = processor.tokenizer.cls_token_id
model.config.pad_token_id = processor.tokenizer.pad_token_id
model.config.vocab_size = model.config.decoder.vocab_size
model.config.eos_token_id = processor.tokenizer.sep_token_id
model.config.max_length = 64
model.config.early_stopping = True
model.config.no_repeat_ngram_size = 3
model.config.length_penalty = 2.0
model.config.num_beams = 4

```

```

In [ ]: # Optimizer
optimizer = AdamW(model.parameters(), lr=5e-5)

```

Training and Evaluation Code

```

In [ ]: # Training Loop
for epoch in range(5):
    model.train() # Set the model to training mode
    train_loss = 0.0 # Initialize training loss accumulator

    for batch in tqdm(train_dataloader, desc=f"Training Epoch {epoch + 1}"):
        # Move batch data to the device
        batch = {k: v.to(device) for k, v in batch.items()}

        # Forward pass
        outputs = model(pixel_values=batch["pixel_values"], labels=batch["labels"])
        loss = outputs.loss

        # Backward pass to calculate gradients
        loss.backward()

        # Step optimizer and zero gradients
        optimizer.step()
        optimizer.zero_grad()

        # Accumulate training loss
        train_loss += loss.item()

    # Print average training loss after each epoch
    print(f"Loss after epoch {epoch}: {train_loss / len(train_dataloader)}")

```

```
Training Epoch 1:  0%|          | 0/1085 [00:00<?, ?it/s]
Loss after epoch 0: 2.5130005232200094
Training Epoch 2:  0%|          | 0/1085 [00:00<?, ?it/s]
Loss after epoch 1: 1.5996478707010295
Training Epoch 3:  0%|          | 0/1085 [00:00<?, ?it/s]
Loss after epoch 2: 1.0869664058981952
Training Epoch 4:  0%|          | 0/1085 [00:00<?, ?it/s]
Loss after epoch 3: 0.9348804777668368
Training Epoch 5:  0%|          | 0/1085 [00:00<?, ?it/s]
Loss after epoch 4: 0.8751201468129312
```



```

In [ ]: # Enhanced Evaluation Loop with Correct Batch Skipping
model.eval() # Set the model to evaluation mode
valid_cer = 0.0 # Initialize CER accumulator
problematic_batches = [] # List to store problematic batches

with torch.no_grad(): # Disable gradient computation during evaluation
    for i, batch in enumerate(tqdm(eval_dataloader, desc="Evaluating")):
        try:
            # Skip specific batches 185 and 186
            if i + 1 in [185, 186]: # Adjust for 1-based indexing in descriptions
                print(f"Skipping batch {i + 1}")
                problematic_batches.append(batch) # Save skipped batch for debugging
                continue

            # Move batch data to the device
            batch = {k: v.to(device) for k, v in batch.items()}

            # Debugging: Check batch content
            print(f"\nProcessing batch {i + 1}")
            print(f"Batch keys: {batch.keys()}")
            print(f"Batch pixel_values shape: {batch['pixel_values'].shape}")
            print(f"Batch labels shape: {batch['labels'].shape}")

            # Ensure batch contains valid data
            if batch["pixel_values"].nelement() == 0 or batch["labels"].nelement() == 0:
                raise ValueError(f"Empty data in batch {i + 1}")

            # Generate predictions
            outputs = model.generate(pixel_values=batch["pixel_values"])
            print(f"Generated outputs shape: {outputs.shape}")

            pred_texts = processor.batch_decode(outputs, skip_special_tokens=True)
            print(f"Predicted texts: {pred_texts[:3]}") # Show first 3 predictions

            # Process labels for comparison
            label_ids = batch["labels"].clone()
            label_ids[label_ids == -100] = processor.tokenizer.pad_token_id
            label_texts = processor.batch_decode(label_ids, skip_special_tokens=True)
            print(f"Label texts: {label_texts[:3]}") # Show first 3 labels

            # Compute CER
            cer = cer_metric.compute(predictions=pred_texts, references=label_texts)
            print(f"Current CER: {cer}")
            valid_cer += cer

        except Exception as e:
            print(f"Error in batch {i + 1}: {e}")
            problematic_batches.append(batch) # Save problematic batch
            continue # Skip the problematic batch and proceed

# Final CER calculation
num_skipped_batches = len(problematic_batches)
if valid_cer > 0:
    avg_cer = valid_cer / (len(eval_dataloader) - num_skipped_batches) # Exclude skipped batches
    print(f"Validation CER after evaluation: {avg_cer:.4f}")
else:
    print("No valid CER calculated due to errors.")

```

Evaluating: 0% | 0/272 [00:00<?, ?it/s]

```
Processing batch 1
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
/usr/local/lib/python3.11/dist-packages/transformers/generation/utils.py:1528: UserWarning: You have modified the pretrained model configuration to control generation. This is a deprecated strategy to control generation and will be removed in v5. Please use and modify the model generation configuration (see https://huggingface.co/docs/transformers/generation\_strategies#default-text-generation-configuration)
  warnings.warn(
```

```
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['- Rückverschlingung aller Staaten', '4. Die ClasssIM Modell inhalisieren', 'GETT & POSst']
Label texts: ['- Gleichberechtigung aller Staaten', '#OneClassSVM Modell initialisieren', 'GET & POST']
Current CER: 0.17216117216117216

Processing batch 2
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['Datenbanken speichern Informationen.', 'In der Pränkal der Bundesspunkt wird', '- sünde unter
te und Teilweise Entlassungen ab QOA']
Label texts: ['Datenbanken speichern Informationen.', 'In der Präambel der Bundesrepublik wird', '- Sinkende Urt
eile und Teilweise Entlassungen ab 1949']
Current CER: 0.14868804664723032

Processing batch 3
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Mutter Juliette verfleucht. Der Streit wird', '5. Regierungsbereitungsbildung', 'reibung von
Nahrung sich auf den Geschmenach']
Label texts: ['Mutter Julietta verflucht. Der Streit wird', '5. Regierungsbeteiligung', 'reibung von Nahrung sic
h auf den Geschmack']
Current CER: 0.11730205278592376

Processing batch 4
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['- Unteilungung der Gruppen:', '- BS passt den Prozess dem Block am', 'die Situation aufgrund
des Imperialismus']
Label texts: ['- Zuteilung der Gruppen:', '- BS passt den Prozess dem Block an', 'die Situation aufgrund des Imp
erialismus']
Current CER: 0.11604095563139932

Processing batch 5
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['Wissenschaftsfunkten Welt der Sprache gibt', 'Das Kängen hüpf über die Wiese.', 'Hypothese f
alsch, Lassenanken 200%']
Label texts: ['Wissentschaftlichen Welt der Sprache gibt', 'Das Känguru hüpf über die Wiese.', 'Hypothese falsc
h, Libanonkrieg 2006']
Current CER: 0.15454545454545454

Processing batch 6
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['1 ... mindestens eine, zwei oder mehr', 'Zustandsauternten beschreiben die Reaktionen eines
Objekts auf verschiedene', 'Ein Schadterling tanzt.']
Label texts: ['1..* .. mindestens eine, zwei oder mehr', 'Zustandsautomaten beschreiben die Reaktionen eines
Objekts auf verschiedene', 'Ein Schmetterling tanzt.']
Current CER: 0.13349514563106796
```

```
Processing batch 7
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Die geht es dir heute?', 'Forderungen die Rückgeordt auf Urtenorder', '- Regelgeleitet']
Label texts: ['Wie geht es dir heute?', 'Forderungen ohne Rückgriffsradt auf Unternehmer', '- Regelgeleitet']
Current CER: 0.18397626112759644

Processing batch 8
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['- Ausgert würweder Mount - Ganst und Umgesoffaktoren', 'Im Gegerung gelangen Will zogenen
in die Dünndarmzelle beein, verhalb es sich um einem', 'Kunsthandweise verbindet Traktionen mit Ihm aften']
Label texts: ['- August wärmster Monat Gunst und Ungunstfaktoren', 'Im Gegenzug gelangen K+-Ionen in die Dünn
darmzelle hinein, weshalb es sich um einen', 'Kunsthandwerk verbindet Tradition mit Innovation.']
Current CER: 0.2844574780058651

Processing batch 9
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['sich Vorstellt z. zieren. Dachum. Das laut -', '- Basisient x nur aus einer Variabelte, neuer
t man das Modell Einzelnde linaue lingsen', 'Lösen linearer Gleichungssysteme']
Label texts: ['sich Vorstellt z.B einen Baum. Das Laut -', '- Besteht x nur aus einer Variable, nennt man das Mo
dell Einfache Lineare Regression', 'Lösen linearer Gleichungssysteme']
Current CER: 0.21148825065274152

Processing batch 10
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['zum Verdrängen. Als er eine Münstime', 'verwendt um sie zu belängen.', '- Stadt vs. Land (agr
enricht-ininstielle Konflikte)']
Label texts: ['zum Verdrängen. Als er eine Männerstimme', 'verwendet um sie zu bekämpfen.', '- Stadt vs. Land (a
grarische-industrielle Konfliktlinie)']
Current CER: 0.15822784810126583

Processing batch 11
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['- Sozial Handeln lässt das aktuelle weitablen', '- hohe Samierungskosten', '- Transnationale
Durchnungen: Über Regierungsinten hinaus (ING')
Label texts: ['- Signal Handler lässt das aktuelle weiterlaufen', '- hohe Sanierungskosten', '- Transnationale B
eziehungen: über Regierungsinstanzen hinaus (NGOs)']
Current CER: 0.1752873563218391

Processing batch 12
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
```

```
Predicted texts: ['nach dem Zweiten Weltkrieg und die Bedeutung der ausgänchen', 'schrieben. Die Reiseberichte h  
ochten den Bahracht', 'nicht vie Seefahrer auch, die ihm Konlgarz']  
Label texts: ['nach dem Zweiten Weltkrieg und die Bedeutung der europäischen', 'schrieben. Die Reiseberichte bra  
chten dem Buchmarkt', 'nicht viele Seefahrer gab, die im Konkurrenz']  
Current CER: 0.22666666666666666
```

```
Processing batch 13  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 31])  
Predicted texts: ['Miss ist ein schönes Land mit voller Natur.', 'Nr.1) abbewenden Stoffwechselbsel = istader St  
off', 'dass sieß, attstehtisch, voll proportionalisiert und zu rötlicher']  
Label texts: ['Mexico ist ein schönes Land mit toller Natur.', 'Nr.1.) abbauendem Stoffwechsel = kataboler Stoff  
wechsel', 'das sie groß, athletisch, wohl proportioniert und zu rötlicher']  
Current CER: 0.23820224719101124
```

```
Processing batch 14  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 35])  
Predicted texts: ['Treatmenten hatten, und allen Personen der Treatment-Gruppen.', 'Wegte Assoziationen bestehen  
zwischen auchten den Klassen?', '- Vorteil: verbändliche Veränderungen desseiten Merkmalsträger festwortfäger']  
Label texts: ['Treatment hatten, und allen Personen der Treatment-Gruppen.', 'Welche Assoziationen bestehen zwis  
chen den Klassen?', '- Vorteile: tatsächliche Veränderungen derselben Merkmalsträger festzustellen']  
Current CER: 0.16966580976863754
```

```
Processing batch 15  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 34])  
Predicted texts: ['Der Durchsinstersiederschlag liegt liegt bei 50mm. Das Diagramm zeigt das', '3. Quasi-Experim  
ent', 'Interpretationsmöglichkeit']  
Label texts: ['Der Durchschnittsniederschlag liegt bei 599mm. Das Diagramm zeigt das', '3. Quasi-Experiment', 'I  
nterpretationsmöglichkeit']  
Current CER: 0.18933333333333333
```

```
Processing batch 16  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 35])  
Predicted texts: ['Auton gehört zu den gegestisch perfekten Menschen und ist gesund. Dage ist Vincent,', 'Mensch  
en kommen müssen, durch kein Anteil', 'Objekte können zur Laufert erzeugt werden']  
Label texts: ['Anton gehört zu den genetisch perfekten Menschen und ist gesund. Dagegen ist Vincent Kurzsichtig,  
, 'Menschen kommen müssen, doch beim Anblick', 'Objekte können zur Laufzeit erzeugt werden.']  
Current CER: 0.1934065934065934
```

```
Processing batch 17  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 30])  
Predicted texts: ['dassen aus, das jedes Kind von Gluut an gewärsen', '2.5 der Normalgleichung ist für große Prob  
leidungen nicht effizient', 'Bestandteile']  
Label texts: ['davon aus, dass jedes Kind von Geburt an gewisse', '(z.B der Normalgleichung) ist für große Probl  
emstellungen nicht effizient', 'Bestandteile']
```

Current CER: 0.19161676646706588

Processing batch 18

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 31])

Predicted texts: ['- Da die deterministisch ist braucht man aus ein ein', 'Glück', 'Datenreplikation sichert Ver
fügbarkeit.']

Label texts: ['- Da sie deterministisch ist braucht man nur ein', 'Glück', 'Datenreplikation sichert Verfügbarke
it.']

Current CER: 0.10882352941176471

Processing batch 19

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 24])

Predicted texts: ['Der Imperialismus', '- Nach 5-jährigen Friedenaborten', 'Das Wasser im Fluss ist einhalt.']

Label texts: ['Der Imperialismus', '- Nach 5-Jährigem Friedenskongress', 'Das Wasser im Fluss ist eiskalt.']

Current CER: 0.12396694214876033

Processing batch 20

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 35])

Predicted texts: ['Hinter der Mann befindet sich eine Palme', '-> Gott kümmert sich nicht genug', '- Transportik
: Def.']

Label texts: ['Hinter dem Mann befindet sich eine Palme', '-> Gott kümmert sich nicht genug', '- Innenpolitik: D
ef.']

Current CER: 0.18253968253968253

Processing batch 21

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 35])

Predicted texts: ['Die Erde Luft so der Nacht.', '- erschönenden Konstanten, welche bestimmte Informationen über
kon das Code bestellten', 'Überblick über das Grignis bekommt. Naterblik']

Label texts: ['Die Eule ruft in der Nacht.', '- vordefinieren Konstanten, welche bestimmte Informationen über Ko
ntext des Codes bereitstellen', 'Überblick über das Ereignis bekommt. Natürlich']

Current CER: 0.2149321266968326

Processing batch 22

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 32])

Predicted texts: ['- Ö konstand, man kann messen wie lange es dauert', '- c) Hauptversammlung', 'Campus-Leben bi
etet vielfältige Mögliche.']

Label texts: ['- v konstant, man kann messen wie lange es dauert', '- c) Hauptversammlung', 'Campus-Leben bietet
vielfältige Möglichkeiten.']

Current CER: 0.14461538461538462

Processing batch 23

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

```
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['- Parameterkode OHG: Produktion = Verkauf von Rund- und- und sichgeigen', '- Erleichterung de
r Einarbeitung in ein Produkt', '- internationalisiert']
Label texts: ['- Rosenstock OHG : Produktion + Verkauf von Rund- und Stichsägen', '- Erleichterung der Einarbeit
ung in ein Produkt', 'internationalisiert']
Current CER: 0.1340782122905028
```

```
Processing batch 24
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['- Recht auf Einigkeit in gespracherten Daten', 'Aurwirkungen der Napoleonischen Herrschaft au
f Europa - 148.8.8-823', '-> das Kolangebaut Störbe und in Wörne']
Label texts: ['- Recht auf Einsicht in gespeicherte Daten', 'Auswirkungen der Napoleonischen Herrschaft auf Euro
pa 17.8.2023', '-> das Kohlenhydrat Stärke und in Wärme']
Current CER: 0.1488095238095238
```

```
Processing batch 25
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- Mere den Unterschied in der Treatment-Gruppe und der', '10, zuschieden lernen unterstützt b
ei der Kriegs', '- Möglichst große Anzahl vom Beobachtungen']
Label texts: ['- Messe den Unterschied in der Treatment-Gruppe und der', 'Maschinelles Lernen unterstützt bei de
r Krebsdiagnose.', '- Möglichst große Anzahl von Beobachtungen']
Current CER: 0.2
```

```
Processing batch 26
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['Der Künfer meht gerne mit helben Farten.', 'Mmen unterschied zu Raterteien & Verbände', '1.11
P: Wissenschiedliche Beobachtung']
Label texts: ['Der Künstler malt gerne mit hellen Farben.', '- MM unterschied zu Parteien & Verbände', '16.1.18:
- Wissenschaftliche Beobachtung']
Current CER: 0.1619718309859155
```

```
Processing batch 27
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['Nährstoffkreislaot', 'Formen der Machtausführung:', 'Dabei spricht das lyrischen. Ich den Lese
r diest']
Label texts: ['Nährstoffkreislauf', 'Formen der Machtausübung:', 'Dabei spricht das lyrische-Ich den Leser direk
t']
Current CER: 0.17681159420289855
```

```
Processing batch 28
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['auf das Thema des Textes uns einztt, In Gotten-', '- Sprache beeinflusst das Denken auch negat
ive', 'Bech umkreist, gibt es war Bewegungen im zweidmensinnen Raum,']
Label texts: ['auf das Thema des Textes anspielt. Im Gegen -', '- Sprache beinflusst das Denken, auch negativ',
```

```
'Bahn umkreist, gibt es nur Bewegungen im zweidimensionalen Raum,']  
Current CER: 0.13592233009708737
```

```
Processing batch 29  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 26])  
Predicted texts: ['- BS zum Tode und viele troßstropfen', 'Protothe - Erstellung', 'beschieden. eigen hat Sehnsu  
cht nach über']  
Label texts: ['- 12 zum Tode und viele Haftstrafen', 'Prototyp-Erstellung', 'beschrieben. Mignon hat Sehnsucht n  
ach ihrer']  
Current CER: 0.22039473684210525
```

```
Processing batch 30  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 32])  
Predicted texts: ['Unterschiedliche Handroreibungen der Erstellung aufgrund von zu', '- Konkehrung von Abstrakte  
n von die Lieferumfang auswisswissenschaft', 'Somittil: Wissenschaft der Zeilensysteme']  
Label texts: ['Unterschiedliche Handhabung der Erstellung aufgrund von z.B:', '- Konkretisierung von Abnahmekrit  
erien und Lieferumfang aus Lastenheft', 'Semiotik = Wissenschaft der Zeichensysteme']  
Current CER: 0.1856763925729443
```

```
Processing batch 31  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 35])  
Predicted texts: ['- kleiner Teil der Population gründet eine neue Population', 'über alles wast machen', '- Str  
uktur der Objekte']  
Label texts: ['- kleiner Teil der Population gründet eine neue Population', 'über alles was wir machen', '- Stru  
ktur der Objekte']  
Current CER: 0.14596273291925466
```

```
Processing batch 32  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 24])  
Predicted texts: ['nicht diesen viel es Gericht zu sehen und', 'Objektidentität', 'Übermorgen ist ein Feiert  
g.']  
Label texts: ['nicht diesem Übel ins Gesicht zu sehen und', 'Objektidentität', 'Übermorgen ist ein Feiertag.']  
Current CER: 0.12083333333333333
```

```
Processing batch 33  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 36])  
Predicted texts: ['Hankenkonkenenenen bühen heitsystemssten.', '- macht sich Sorgen um ihren Ruf', 'Trennung von  
Formation & Inhalt']  
Label texts: ['Hacker knacken Sicherheitssysteme.', '- macht sich Sorgen um ihren Ruf', 'Trennung von Formatieru  
ng & Inhalt']  
Current CER: 0.2277580071174377
```

```
Processing batch 34  
Batch keys: dict_keys(['pixel_values', 'labels'])
```



```
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 43])
Predicted texts: ['- Politisberatung als Beobachtung zweither Ordnung hötte', '- Einzeltenstellungszezifikation laut Plan', '- Anpassung / Vermeidung']
Label texts: ['- Politikberatung als Beobachtung zweiter Ordnung hätte also die', '- Einzeltestfallspezifikation laut Plan', '- Anpassung / Vermeidung']
Current CER: 0.2199074074074074
```

```
Processing batch 35
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 22])
Predicted texts: ['0 ... "haut keine, eine, die Zwei oder mehr', '2. Nomierende Macht', 'zu schließen']
Label texts: ['0..* .. hat keine, eine, zwei oder mehr', '2. Nomierende Macht', 'zu schließen']
Current CER: 0.11764705882352941
```

```
Processing batch 36
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 20])
Predicted texts: ['Einsatzbereich Sequenz-Diagramm', 'der Vergleitet die Gruppe.', 'Künstliche Intelligenz lernt von Daten.']
Label texts: ['Einsatzbereich Sequenz-Diagramm', 'oder Vergleich der Gruppe.', 'Künstliche Intelligenz lernt von Daten.']
Current CER: 0.12109375
```

```
Processing batch 37
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['- Jahrechwahrhtenfahrts vom negativ 14-30 Grad', 'Hypothesen:', '- Datenerheidung']
Label texts: ['- Jahresdurchschnittstemperatur 21-30 Grad', 'Hypothesen:', '- Datenerhebung']
Current CER: 0.19622641509433963
```

```
Processing batch 38
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Proteien des Lebensmedläre Natur', '- ungleiche Verteilung von vertvolten Ressourcen /', 'Beschreibung der Fägleiten des Softwareprodukte anhand von Anwendungsgefüren:']
Label texts: ['Parteien als intermediärer Akteur', '- ungleiche Verteilung von wertvollen Ressourcen /', 'Beschreibung der Fähigkeiten des Softwareprodukts anhand von Anwendungsfällen']
Current CER: 0.20212765957446807
```

```
Processing batch 39
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['löskte Calwandient. Das Wasser sind durch zuach zur Wasserstränge gebte', '- Zustimmung der Signierte', 'Ursachen:']
Label texts: ['lösliche Calciumcarbonat. Das Wasser wird danach zur Wasserstrahlpumpe geleitet', '- Zustimmung der Siegermächte', 'Ursachen:']
Current CER: 0.234375
```

```
Processing batch 40
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['4 Vorlusage', 'Studenten erkunden neue Hobbys und Interessen.', '18% des Wissenverbrauch wird
überbasisch benutzt']
Label texts: ['# Vorhersage', 'Studenten erkunden neue Hobbys und Interessen.', '18% des Wasserverbrauchs wird i
nnerhäuslich benutzt']
Current CER: 0.1949685534591195
```

```
Processing batch 41
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['eine Rachle', '- Schade ziehre Auswertung möglich', 'Zusammen: - Sicht der Erde: Rachte kommt
nach 10 Jahren zurück;']
Label texts: ['eine Rede.', '- Schnelle erste Auswertung möglich', 'Zusammen: Sicht der Erde: Rakete kommt nac
h 10 Jahren zurück;']
Current CER: 0.1811320754716981
```

```
Processing batch 42
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['- eigensändig Softwarebraute mit können Schnittständen', 'durch Messung ist numerisches Relat
iv getrenacht', '- Ressourcenausanhaltung und Lebensbedingen sind']
Label texts: ['- eigenständige Softwarebausteine mit klaren Schnittstellen', 'durch Messung ins numerische Relat
iv gebracht', '- Ressourcenausstattung und Lebensbedingungen sin']
Current CER: 0.17717717717717718
```

```
Processing batch 43
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 40])
Predicted texts: ['- Momentum verwendet eine konstantante Lernrate für als Festane -> umgeötig', 'in Bereich vür
de, Wissenschaft', 'Voraussetzung für einfachte Änderungen mit alter hatte']
Label texts: ['- Momentum verwendet eine konstante Lernrate für alle Features -> ungünstig', 'im Bereich Kirche,
Wissentschaft', '- Vorraussetzung für einfache Anderungen mit alter table']
Current CER: 0.189010989010989
```

```
Processing batch 44
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['Bei den unter 25 jährigen sind 17,8% arbeitsliches', '- Fürte Reizionsrechte, Glünstellung de
r', 'Intention : 1972 hat man sie gereichnet']
Label texts: ['Bei den unter 25 jährigen sind 13,8% arbeitslos', '- Feste Religionsrechte, Gleichstellung der',
'Intention: Wozu hat man sie gezeichnet']
Current CER: 0.15335463258785942
```

```
Processing batch 45
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
```

```
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 40])
Predicted texts: ['- Repräsentation', 'fordert ihn auf die verliehrende Zeit zu', 'Rechte']
Label texts: ['Repräsentation', 'fordert ihn auf die verbleibende Zeit zu', 'Rechtecke']
Current CER: 0.2507462686567164
```

```
Processing batch 46
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['Ein Vogel sitzt in einem hohen Baum.', 'Führungsaufgaben', '- große Umverteilung -> Gleichheit der Leistungen']
Label texts: ['Ein Vogel sitzt in einem hohen Baum.', 'Führungsaufgaben', '- große Umverteilung => Gleichheit der Leistungen']
Current CER: 0.11598746081504702
```

```
Processing batch 47
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['- Schließung der Macht über, soziale Schließungsprozesse', 'Die Durchsichtweder belegt 18,71 und der Jahrsniederschlag 677mm', '- Selbstbewusstsein']
Label texts: ['Stabilisierung der Macht über soziale Schließungsprozesse', 'Die Durchschnittstemperatur beträgt 17,1°C und der Jahresniederschlag 677mm.', '- Selbstbewusstsein']
Current CER: 0.18133333333333335
```

```
Processing batch 48
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
Predicted texts: ['Rangflaggenene Gesen römischen die Kartoffeländer, abgehen ihren sie ihre über', '- Nachtor, viele Ausprägungen des Faktors', 'Grazenlisierung anpekt']
Label texts: ['Rangflaggenene Gesen römischen die Kartoffeländer, abgehen ihren sie ihre über', '- Nachtor, viele Ausprägungen des Faktors', 'Grazenlisierung anpekt']
Current CER: 0.20417633410672853
```

```
Processing batch 49
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['und weiß das Newton und Einstein Gebirgsent sind.', 'Bestimme die Lernrate nach dem Trainings- und dem Testdatensatz', '4%. Menschen werden geboren und sterben auf natürlichem Weg.']
Label texts: ['und weiß das Newton und Einstein Geheimagenten sind.', 'Bestimme die Lernrate nach dem Trainings- und dem Testdatensatz', '9. Menschen wurden geboren und sterben auf natürlichem Wege.']
Current CER: 0.09798270893371758
```

```
Processing batch 50
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['perkonwissen & notwendulen', '- Zahl der Personen in erwerbsfähigen Alter sinkt, da es zu einer', 'Zudem kritisiert er Pinker, damit er glaubwürdig wirkt. Jedoch betont er das']
Label texts: ['reproduzieren & instrumentalisieren', '- Zahl der Personen in erwerbsfähigen Alter sinkt, da es zu einer', 'Zudem kritisiert er Pinker, damit er glaubwürdig wirkt. Jedoch betont er das']
```

Current CER: 0.23008849557522124

Processing batch 51

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 37])

Predicted texts: ['- Tagefänkte Wissar über ühnchen sich in allen Sprachen', 'Das Buch wird bald veröffentlicht.', '3.) Der Transporten ist ein passiert immer Transportenismus, weit Energiegie Energiegie in Person von ATP']

Label texts: ['- Tagtägliche Wörter ähneln sich in allen Sprachen', 'Das Buch wird bald veröffentlicht.', '3. Der Transport ist ein primär aktiver Transportmechanismus, weil Energie in Form von ATP']

Current CER: 0.2048780487804878

Processing batch 52

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 31])

Predicted texts: ['Campus-Rechts bietet studentische Medien.', '- Theorien untersstiedliche Tiefe', '2. Chrommst y und die Universalgrammmatik']

Label texts: ['Campus-Radio bietet studentische Medien.', '- Theorien unterschiedliche Tiefe', '2. Chomsky und die Universalgrammatik']

Current CER: 0.16030534351145037

Processing batch 53

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 42])

Predicted texts: ['Nralle Attribute müssen attonax sein, z.B vor 8. Nachraum', 'Stärkenenenen', 'Nr.1.) Hochwasser Unter einen Hochnasser Wortteil man das Anstiegen des Aussen des Wasserstander']

Label texts: ['1.NF alle Attribute müssen atomar sein, z.B. Vor- & Nachname', '- Stärken - Schwächen', 'Nr.1.) Hochwasser: Unter einem Hochwasser versteht man das Ansteigen des Wasserstandes eines']

Current CER: 0.2634989200863931

Processing batch 54

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 44])

Predicted texts: ['- Refne, abgegaute Verbindung', 'Die (kaufzeit-Komplexität der Beschmung der Normalung hängt aus den Prozahl der Frechurs aus ziel)', '- Verreinfnung unwendig']

Label texts: ['- Refine, abgeleitete Verbindung', '-> Die (Laufzeit-Komplexität der) Berechnung der Normalgleichung hängt von der Anzahl der Features ab!', '- Vereinfachung notwendig']

Current CER: 0.18991097922848665

Processing batch 55

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 36])

Predicted texts: ['- Glaube an die Legität der Verfahrenswissen', 'Somit hatt das Eins auch eine Kontrollfunkten.', '- Interviewe Logik']

Label texts: ['- Glaube an die Legalität der Verfahrensweisen', 'Somit hat das EP auch eine Kontrollfunktion.', '- Interne Logik']

Current CER: 0.11711711711711711

Processing batch 56

```
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['An welchern Fluss führten die römischen Kaiser immer Feldtuge durch?', 'dass nur die rechte H
interhätte entwickelt ist. Da die Länkte Hälfte', 'Siele 1. Postoldt bz seine ausgegebauete Definition:']
Label texts: ['An welchem Fluss führten die römischen Kaiser immer wieder Feldzüge durch?', 'dass nur die rechte
Hirnhälfte entwickelt ist. Da die linke Hälfte', 'Sein 1. Postulat bzw. seine ausgehende Definition:']
Current CER: 0.21868365180467092
```

```
Processing batch 57
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Gewalt gegen die die nicht arbeiten', 'Der Ort Segen liegt in China, also ein', 'Zwei oder me
hrststufig geschichtet zum an Merkmalsträge']
Label texts: ['- Gewalt gegen die, die nicht arbeiteten', 'Der Ort Sezuan liegt in China, also ein', 'Zwei oder
mehrstufig geschichtet = um an Merkmalsträger']
Current CER: 0.15902140672782875
```

```
Processing batch 58
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 49])
Predicted texts: ['ein Datenslüssels: dass der Tabelte & steht mit 0. In. D.n. Dakreiben aus dem Tabelle für ein
Beziehung', 'Lösungen = Produkt', 'Leit daraute nicht lange. Auf dem ofteren Tabelt zu es keinen']
Label texts: ['- ein Datensatzsatz aus der Tabelle B steht mit 0..m Datensätzen aus der Tabelle A in Beziehung.'
, 'Lösungsweg = Produkt', 'heit dauerte nicht lange. Auf den offenen Feld kam es einen']
Current CER: 0.300531914893617
```

```
Processing batch 59
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 37])
Predicted texts: ['Kriegsprojende:', 'leichte Änderung von Datenstrukturen (Kritzte) innerhalb einer Klasse eine
den Besteiligung', 'Entdeckungen Grenzen unser Verständnis der Welt.']
Label texts: ['Kriegspropaganda:', '- Leichte Änderung von Datenstrukturen (Attribute) innerhalb einer Klasse oh
ne Beteiligung', 'Entdeckungen formen unser Verständnis der Welt.']
Current CER: 0.15526315789473685
```

```
Processing batch 60
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['kann geleit werden und dann von Gefahren oder mössischen aufgebren wurde.', '-> Dahtoriehle E
rger, Rahmenwaffler, Hingze, Ranger...', '- Einfunker der sozialen Machtmirtschaft']
Label texts: ['kennen gelernt wurden und dann von Seefahrern oder Mitreisenden aufgeschrieben wurde.', '-> Bakte
rielle Erreger, Flammenwerfer, Flugzeuge, Panzer...', '- Einfuhr der sozialen Marktwirtschaft']
Current CER: 0.20334928229665072
```

```
Processing batch 61
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
```

```
Predicted texts: ['Anderbrauen, die Anwester der Software aufgrund einer Arbeitshänge', 'Theater bringt Emotionen zum Ausdruck.', 'Sicht einer Demonstantie.']
Label texts: ['Anforderungen die der Anwender der Software aufgrund seiner Arbeitsabläufe hat', 'Theater bringt Emotionen zum Ausdruck.', 'Sicht einer Demonstrantin.']
Current CER: 0.1337579617834395
```

```
Processing batch 62
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Ihn nicht etma der Hals mit abgeschritten mürsche Einklich', '- DNA Polymise baut neu Folgestung', 'Fakt und Klasse']
Label texts: ['ihn nicht etwa der Hals mit abgeschnitten würde. Endlich', '- DNA Poylmerase baut neuen Folgestrang', 'Paket und Klasse']
Current CER: 0.19292604501607716
```

```
Processing batch 63
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['3. Betriebe verlaufen Ereignisse', 'Ich erre gerne Pizza mit viel Küre.', '- fachsühre, aufgebewerientiert']
Label texts: ['3. Betriebe verkaufen Erzeugnisse', 'Ich esse gerne Pizza mit viel Käse.', '- fachliche, aufgabenorientiert - Strukturierungskonzept']
Current CER: 0.19601328903654486
```

```
Processing batch 64
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Entwaußprozess', 'neue Kapital', 'Beispielungen betren Eineoffe, um einen direkten Beziehung muss eigen Handeln']
Label texts: ['Entwurfprozess', 'neue Kapital', 'Beispielierzählungen betonen Einzelfälle, um einen direkten Bezug zum eigenen Handeln']
Current CER: 0.20110192837465565
```

```
Processing batch 65
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 55])
Predicted texts: ['Güstenhängig bieten Einblicke in aktuelle Forschung.', 'miteinander Verknüpft. Durch diese', 'Antiport handelt.']
Label texts: ['Gastvorträge bieten Einblicke in aktuelle Forschung.', 'miteinander Verknüpft. Durch diese', 'Antiport handelt.']
Current CER: 0.3310580204778157
```

```
Processing batch 66
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 22])
Predicted texts: ['- Trammen, Dahl', 'Auf dem zweiten Bild sind 4 Menschen zu', '- Beschreibung der Enteressen Validität']
Label texts: ['- Truman, Dahl', 'Auf dem zweitem Bild sind 4 Menschen zu', '- Bedrohung der internen Validität']
Current CER: 0.17537313432835822
```

```
Processing batch 67
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['Die sind Gewinn bekommen Insten und Besta?', 'Fremdschlüssel', 'Politische Kommunikation']
Label texts: ['Wie viel Gewinn bekommen Anton und Berta?', 'Fremdschlüssel', 'Politische Kommunikation']
Current CER: 0.15

Processing batch 68
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['- Designation durch durch die Gefaltigkeit', '5) Die Erstehung und der Lebenschluss von Parte
ien', '- Evolution der Geschlechter']
Label texts: ['- Designation durch die Gefolgschaft', '5.) Die Entstehung und der Lebenszyklus von Parteien', '-
Evolution der Geschlechter']
Current CER: 0.18618618618618618

Processing batch 69
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 45])
Predicted texts: ['- Blöche können nicht im Lußend gewändert', '- Die Regierung sozten "Gesprücht"', 'Unterschie
de und Industrielität der Menschen']
Label texts: ['- Blöcke können nicht im Laufend geändert', '- Die Regierung sei dazu "gesprächsbereit"', 'Unters
chiede und Individualität der Menschen']
Current CER: 0.2531969309462916

Processing batch 70
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Systemstest', 'Wiehremert man Kinder, die jahlung isoliert gelebt?', 'Selbstbehmung']
Label texts: ['Systemstest', 'Wie nennt man Kinder, die jahrelang isoliert gelebt haben?', 'Selbstbehauptung']
Current CER: 0.2052980132450331

Processing batch 71
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['- Keine regionale soklärung -> Bewusstsein', '-> soziale Herkunft', 'Der Oderart ist heute ha
nde.')]
Label texts: ['- Keine rationale Erklärung -> Bewusstsein', '-> soziale Herkunft', 'Der Oberarzt ist heute kran
k.')]
Current CER: 0.10097719869706841

Processing batch 72
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['Der Text hat einen Hunderit im Bereich als Sachen. Es werden erste Grundhän', '-> welche Pers
onen führen diese Hutgabe zur Zeit durch', 'Anaktische Schnitte']
```

```
Label texts: ['Der Text hat einen Hakenstil im Bereich des Satzbaus. Es werden extra Grammatikalische', '-> welc  
he Personen führen diese Aufgaben zur Zeit durch', 'Analytische Schritte']  
Current CER: 0.20307692307692307
```

```
Processing batch 73  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 34])  
Predicted texts: ['Das Baby sprach sich seine ersten Wörter.', 'danteil', 'Chromen besteht aus zwei investisch  
en DNA-Frischer Chromoshen) + Diese werden von']  
Label texts: ['Das Baby sprach soeben seine ersten Wörter.', 'dunkel', 'Chromosom besteht aus zwei identischen D  
NA-Fäden (Chromatiden). Diese werden am']  
Current CER: 0.2
```

```
Processing batch 74  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 32])  
Predicted texts: ['Gründe für die Veränderung des Inspaktor:', '- Sprachkennenens durchs Luftold, Eltern, Frank',  
'- Durchs']  
Label texts: ['Gründe für die Veränderung des Inspektors:', '- Sprachlernen durchs Umfeld, Eltern, Freunde', '-  
Budgets']  
Current CER: 0.20221606648199447
```

```
Processing batch 75  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 31])  
Predicted texts: ['- Dieitene gute Use für Staatsstruktsische Politiktenitung', '- Ressourcenansantung und Lebens  
bedingungen sind', 'des Eichts verlässt den LASER durch einem Indokwortllassen Sogeel.']  
Label texts: ['- Weitere gute Idee für konstruktivistische Politikberatung', '- Ressourcenausstattung und Lebens  
bedingungen sind', 'des Lichts verlässt den LASER durch einen halbdurchlässigen Spiegel.']  
Current CER: 0.1962025316455696
```

```
Processing batch 76  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 30])  
Predicted texts: ['Jedes Objekt aus A steht mit mindesten 1 liegt aus 3 in Sozialnung', 'Erlang ist für, verteil  
te Systeme', '-> Gott ist zu reht und wird ausgesetzt']  
Label texts: ['Jedes Objekt aus A steht mit mindestens 1 Objekt aus B in Beziehung', 'Erlang ist für verteilte S  
ysteme.', '-> Gott ist zu nett und wird ausgenutzt']  
Current CER: 0.1566265060240964
```

```
Processing batch 77  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 42])  
Predicted texts: ['- Primänktion, Herstellung von öffentlichkeiten (z. Meinungsbildungfunktung)', 'Organgramm',  
'- Abhängigkeiten erfunkten in diese Restung von den Kongeben zu den Verhaltenenenen Konstrukten']  
Label texts: ['- Primärfunktion: Herstellung von Öffentlichkeiten (=> Meinungsbildungsfunktion)', 'Organigramm R  
osenstock', '- Abhängigkeitsbeziehungen verlaufen immer in eine Richtung von den konkreten zu den abstrakten Mod  
ulen']  
Current CER: 0.21447721179624665
```



```
Processing batch 78
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 39])
Predicted texts: ['- Nutterangräuft ist nur beschränkte Personenpunkt', '- gehössen nicht zum konkreten Objekt ei
ner Klasse, sondern Klasse zt', '- Häufige Bennttrag von Farten']
Label texts: ['- Datenzugriff ist nur berechtigigten Personen erlaubt', '- gehören nicht zum konkreten Objekt eine
r Klasse, sondern Klasse selbst', '- Häufige Benutzung von Farben']
Current CER: 0.19951923076923078
```

```
Processing batch 79
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['- Beschreibung: Erführungen eines Teilsting eines Strönges', 'Der Lärm stört meine Konzentrati
on.', '- Spalkonten']
Label texts: ['- Beschreibung: Extrahieren eines Teilstrings eines String', 'Der Lärm stört meine Konzentration.
', '- Spaltentypen']
Current CER: 0.18322981366459629
```

```
Processing batch 80
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Aufbaut', '- Problenkösung T gas und mögliche', 'erschaffen, die das Leben das zweiten Gesell
en']
Label texts: ['Aufbau:', '- Problemlösung Tipps und mögliche', 'erschaffen, die das Leben des zweiten Gesellen']
Current CER: 0.11295681063122924
```

```
Processing batch 81
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['- Pflgliche Ursache verwnsucht durch die Mimmstime', 'Mein Nachbar hat sich ein neues Auto ge
kauft.', '- Der Ränsturbegen im Schalten des Daraus']
Label texts: ['- Plötzliche Unruhe verursacht durch die Männerstimme', 'Mein Nachbar hat sich ein neues Auto gek
auft.', '- Der Römertorbogen im Schatten des Doms']
Current CER: 0.17117117117117117
```

```
Processing batch 82
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['Abstände zwischen den Objektion in der Rangfolge', '- Öklingsische Wiese', '- Interesive Kont
nis über Tälle']
Label texts: ['Abstände zwischen den Objekten in der Rangfolge', 'Ökologische Inferenz', '- Intensive Kenntnis ü
ber Fälle']
Current CER: 0.21068249258160238
```

```
Processing batch 83
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
```

```
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['Emer immer ein gefragt dann,', 'Geschäftsleitung', 'Texten: Nachbar Wert dann sich verändern']
Label texts: ['Er war immer ein gefragter Mann,', 'Geschäftsleitung', '- Problem: wahrer Wert kann sich verändern']
Current CER: 0.2229965156794425
```

```
Processing batch 84
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['1. Privat eigenenten', '- überwus sowende Machtumsältnis: Autorität', '- Hohe von Wänner. Für st lerfen']
Label texts: ['1. Privateigentum', '=> daraus resultierende Machtverhältnis: Autorität', '- Goethe von Weimarer Fürst berufen']
Current CER: 0.20132013201320131
```

```
Processing batch 85
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['Das Auto führt auf der Autobahn.', 'Fruckreich und Russland', 'Eigenschaften von Hypothese n.']
Label texts: ['Das Auto fährt auf der Autobahn.', 'Frankreich und Russland', '- Eigenschaften von Hypothesen.']
Current CER: 0.2277580071174377
```

```
Processing batch 86
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Kreativitätsterten', '- Deduktives Vorgehen ihr schleiten wissensstand nicht', '2. Funktion: Definitioniert implementiert einen Funktionsbezug des politischen Systemen Systemen Systemss']
Label texts: ['Kreativitätstechniken', 'Deduktivies Vorgehen bei schlechtem wissensstand nicht', '2. Funktion: Definition impliziert einen Funktionsbezug des politischen Systems']
Current CER: 0.1903485254691689
```

```
Processing batch 87
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['- Einkommen ohren Bedarfprüfung', '- Metapfaher (N.3) -> Sachen überzeit zu Ende gehen', 'Die Physisor essen Äbernflassen und Mülius teilt']
Label texts: ['- Einkommen ohne Bedarfsprüfung', '- Metapher (V.8) -> Es kann jederzeit zu Ende gehen', 'Die drei Physiker essen Abendessen und Möbius teilt']
Current CER: 0.21065989847715735
```

```
Processing batch 88
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['die Subproben', 'Der Stunden denhalten Diesen nehmen haben die magnetischen Seen und Flüsse', '- Kompen auswahl']
Label texts: ['der Stichproben', 'paar Stunden abschalten. Dagegen machen die ausgetrockneten Seen und Flüsse', '- Klumpenauswahl']
Current CER: 0.2680851063829787
```

```
Processing batch 89
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['verarbeiteten Rohstoffte weiter verarbeiten zu Produktionsmittel und diese werden', '- schönk
er oder schlechter Ort', 'Vorstellen und Gescheckt müssen et']
Label texts: ['verarbeiteten Rohstoffe weiter verarbeitet zu Produktionsmittel und diese werden', '- schöner ode
r schlechter Ort', 'Vorsilben und Geschlecht müssen erlernt']
Current CER: 0.22055137844611528
```

```
Processing batch 90
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['Gehalteterstehung -> Vorstand', 'Litzt vordrängt tinglich etwas, nur von', '- Herstellung vo
n Sprangstoffen']
Label texts: ['Gehaltsfestsetzung -> Vorstand', 'selbst verdränge täglich etwas, nur um', '- Herstellung von Spr
engstoffen']
Current CER: 0.1815068493150685
```

```
Processing batch 91
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['Ein Gedanke flugt.', '-> Ökologie, Ökonomie und sozialer (Kulturelus)', '- Einheiteittliche A
PI']
Label texts: ['Ein Gedanke fliegt.', '(-> Ökologie, Ökonomie und soziales / kulturelles)', '- Einheitliche API']
Current CER: 0.13372093023255813
```

```
Processing batch 92
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['nach hötze z.B. in der Kirche. Die Menschen frochen viel in der Wissenschaft', 'Die Reakterze
it begrenzt die Frequenz der Aktionspotentialte. Hat eine Nervenle', 'Ginde ich nicht, da, da man man muss sich
nie gelent hat oder man auch nicht']
Label texts: ['nach hilfe z.B. in der Kirche. Die Menschen forschen viel in der Wissenschaft', 'Die Refraktärzei
t begrenzt die Frequenz der Aktionspotentialte. Hat eine Nervenzelle', 'finde ich nicht gut, da man sowas nie gel
ernt hat aber man auch nicht']
Current CER: 0.16780045351473924
```

```
Processing batch 93
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Berechnung des Zinsalzes:', '- erste Vorsteilungen beim gegenüber schren', '- Wiederholung']
Label texts: ['Berechnung des Zinssatzes:', '- erste Vorstellungen beim gegenüber stehen', '- Wiederholung']
Current CER: 0.16610169491525423
```

```
Processing batch 94
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
```

```
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['Best kann aus normalen muss zwert exportiert werden, wird auch aus auch gegeordt.', '-> wird
für laut-wel. Erdeiltern, beliedet', '- Durch die konztrakraft dreht sich der Elektromor']
Label texts: ['Box kann aus normalen Fluß entfernt werden, wird dann aber auch ignoriert.', 'Go ist für Cloud- u
nd Back-End-Entwicklung beliebt.', '- Durch die Lorentzkraft dreht sich der Elektromotor']
Current CER: 0.2826086956521739
```

```
Processing batch 95
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['Grot ohne Streck diese dassen kommt, durch die', 'des Erloren und Denken stark', '-> geno- un
d phänstigische Unterscheidungen']
Label texts: ['Graf ohne Strafe davon kommt, obwohl die', 'das Erfahren und Denken stark', '-> geno- und phänoty
pische Unterscheidungen']
Current CER: 0.22108843537414966
```

```
Processing batch 96
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['- Diese Teilt man dann in Gruppen ein zu Bracht Vorbeilung.', '- Die Berechnung der Normalgab
altung eigert sich für komplexe Problemeungen', 'Der Mann kann nicht gut beihen.']
Label texts: ['- Diese Teilt man dann in Gruppen ein z.B nach Verlag', '- Die Berechnung der Normalgleichung eig
net sich nicht für komplexe Problemstellungen', 'Der Mann kann nicht gut kochen.']
Current CER: 0.16161616161616163
```

```
Processing batch 97
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- das Vertel aller Seaktionen wegen versatz Temperature (Z.153- 164)', 'Die Wahlwirtsietzte b
ietet eine Menge Bücher.', '- Personen finden sich nicht in angebotenen Kategorien wieder']
Label texts: ['- drei Viertel aller Sanktionen wegen verpasster Termine (Z.153-154)', 'Die Schulbibliothek biete
t eine Menge Bücher.', '- Personen finden sich nicht in angebotenen Kategorien wieder.']
Current CER: 0.1820388349514563
```

```
Processing batch 98
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 45])
Predicted texts: ['Wegen den Klängen und nicht der Bedeutung', 'Modelle angengrenze (Klassen, passablenenenen Nus
schreibungen) in relationalen Modellbild mit Integrationen Vergebildungen', 'grondenden sind Gegen. Erde sinde R
ade spreicht']
Label texts: ['Wegen den Klängen und nicht der Bedeutung', '- Modellierungskonzepte (Klassen, Assoziationen, Ver
erbung) im relationalen Modell mit Integritätsbedingungen verbinden', 'vorhanden sind. Gegen Ende seiner Rede sp
richt']
Current CER: 0.2826552462526767
```

```
Processing batch 99
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Seiten unter 70 Prozent', 'Humpfestadt Lande', '... ein Konzept, das auf der gegen Welt unter
```

```
reitet ist.']  
Label texts: ['selten unter 70 Prozent', 'Hauptstadt Luanda', '... ein Konzept, das auf der ganzen Welt verbreit  
et ist.']  
Current CER: 0.09266409266409266
```

```
Processing batch 100  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 30])  
Predicted texts: ['-> hüpfte den "0 Problem', 'Batch-Gorfe in Modell für, C4,4, y, batchisse=20, ...gaben', 'Die  
Ausgänung von Flächen für die garantwirtschaftliche Nutzum']  
Label texts: ['-> Löst das I-0 Problem', 'Batch-Größe in model.fit(X, y, batch_size=20, ...) angeben', 'Die Anei  
gnung von Flächen für die agrarwirtschaftliche Nutzung']  
Current CER: 0.22093023255813954
```

```
Processing batch 101  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 40])  
Predicted texts: ['- Mögliche Probleme bei Frage verstehen?', '1. Berechne eine Schätzung für W', '- Konkret geg  
, damit im Rahmen man']  
Label texts: ['Mögliche Probleme bei "Frage verstehen":', '1. Berechne eine Schätzung für w', '- Konkret genug,  
damit im Rahmen machbar']  
Current CER: 0.23076923076923078
```

```
Processing batch 102  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 34])  
Predicted texts: ['Zusammenfassend lässt sich sich spegen, dass man mit Hilfe von sozialwissern', 'Studentenen s  
chätzen den Vielfalt an Klassen.', '- Im Jahr 2005 wissen Frankweich und Südstoffe wichtige Handlunger von']  
Label texts: ['Zusammenfassend lässt sich sagen, dass man mit Hilfe von sozialwissen-', 'Studenten schätzen die  
Vielfalt an Kursen.', '- Im Jahr 2000 waren Frankreich und Süd-Afrika wichtige Handelspartner von']  
Current CER: 0.1658653846153846
```

```
Processing batch 103  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 28])  
Predicted texts: ['- Dadurch entsetzt das Problem des Overführung', 'Selbstreflexien', '- Institutionen, Konfigu  
ration, Besteutung oder Ausführung von Informationen']  
Label texts: ['- Dadurch entsteht das Problem des Overfittings', 'Selbstreflexion', '- Installation, Konfigurati  
on, Bereitstellung oder Ausführung von Informationseinheiten']  
Current CER: 0.14775725593667546
```

```
Processing batch 104  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 44])  
Predicted texts: ['Außerformenierung bedeut, dass das Kapital den Unternehmen', '- häuft das Quantum ab -> Unter  
hreibung', 'Massen besuchenhalte Schütze.']  
Label texts: ['Außenfinanzierung bedeutet, dass das Kapital dem Unternehmen', '- Läuft das Quantum ab -> Unterbr  
echung', 'Museen bewahren kulturelle Schätze.']  
Current CER: 0.22189349112426035
```

```
Processing batch 105
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['Kabarett bringt Humor in den Alltag.', 'indischen und der gleichheit aller Menschen', 'Das Ab
rd galoert über die Wiese.']
Label texts: ['Kabarett bringt Humor in den Alltag.', 'irdischem und der Gleichheit aller Menschen', 'Das Pferd
galoppiert über die Wiese.']
Current CER: 0.1488673139158576
```

```
Processing batch 106
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- Befragte müssen zusammen oder absteigen in Abstrafungen.', '5% verspricht unterschiede und
zuertlöse Kommentität.', 'KI optimiert Energieeffizierz.']
Label texts: ['- Befragte müssen zustimmen oder ablehnen in Abstufungen.', '5G verspricht ultraschnelle und zuve
rlässige Konnektivität.', 'KI optimiert Energieeffizienz.']
Current CER: 0.15176151761517614
```

```
Processing batch 107
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['- Parteien: Aggregation gesall schafflicher Interessen', 'Nationalstaats am Besonden Seiten s
tarbe', 'Konferenzen ermögkeitenden beschanchen im Högerten.']
Label texts: ['- Parteien: Aggregation gesellschaftlicher Interessen', 'Nationalstaats an. Besonders Serbien str
ebte', 'Konferenzen ermöglichen den Austausch mit Experten.']
Current CER: 0.1846590909090909
```

```
Processing batch 108
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- In jedem Anten, des Entscheidungsstaum muss bestimmt werden, wie die Datenpunkte', '- Diese
r Abstand & soll maximiert werden', '- Kausche Anzessen oder mittle von theoretischen Erwegungen möglich']
Label texts: ['- In jedem Knoten des Entscheidungsbaum muss bestimmt werden, wie die Datenpunkte', '- Dieser Abs
tand S soll maximiert werden', '- Kausale Aussagen nur mithilfe von theoretischen Erwägungen möglich']
Current CER: 0.11463414634146342
```

```
Processing batch 109
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
Predicted texts: ['- Beobachtung von unterschiedlichen Zuständen führt zur', 'in elektrede Energie soziale in Wä
rme umgewandelt', 'In Wissensheit waren diese Offizwasser aber nicht erweit das die Portugiesen die wenn']
Label texts: ['- Beobachtung von unterschiedlichen Zuständen führt zur', 'in elektrische Energie sowie in Wärme
umgewandelt', 'In Wirklichkeit waren die Afrikaner aber nicht erfreut das die Portugiesen da waren']
Current CER: 0.23484848484848486
```

```
Processing batch 110
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
```

```
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 37])
Predicted texts: ['angehr, dass die Bluteppenpunkten sich schneller erbeit als die Räuberpopulation', 'die Size,
die Tester mit ihren unwesentlichen Daten und den Navigation-', '- Likart-Skala']
Label texts: ['aussagt, das die Beutepopulation sich schneller erholt als die Räuberpopulation', 'eine Skizze, d
ie die Fenster mit ihren wesentlichen Daten und den Navigations -', '- Likert-Skala']
Current CER: 0.22604422604422605
```

```
Processing batch 111
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['Der Mensch Jaht zum arbeiten, wird verbessern um zu zuarten', '- Umplementierung und Überführu
ng', 'stehtzbische Auswertungen zu können']
Label texts: ['Der Mensch lebt zum arbeiten, wird geboren um zu sterben.', '- Implementierung und Überführung',
'statistische Auswertungen machen zu können']
Current CER: 0.14106583072100312
```

```
Processing batch 112
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['Die Brücke überpannt den Fluss.', 'Forderung gibt einen Verfang zwischen Unternehmen und Fact
or', 'Der Eisankern konzentriert Verstürtlichkeit das Magnetfeld und angest ingefür,']
Label texts: ['Die Brücke überspannt den Fluss.', 'Factoring setzt einen Vertrag zwischen Unternehmen und Factor
', 'Der Eisenkern konzentriert / verstärkt das Magnetfeld und sorgt insb. dafür,']
Current CER: 0.20054945054945056
```

```
Processing batch 113
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 41])
Predicted texts: ['- je niedeiner das Modell innerhalb der Hierarchie gegen sozialistischen Problems liet es',
'- Zuk: Überpraltung von Theorie an einem einem wichtig zu', '- Herrschaft soll heißen die Chance, für einen Ber
ehl bestimmten Inhalts']
Label texts: ['- Je niedriger das Modul innerhalb der Hierarchie, desto spezifischere Probleme löst es', 'Ziel:
Überprüfung von Theorie an einem "wichtigen Fall"', '- "Herrschaft soll heißen die Chance, für einen Befehl best
immten Inhalts']
Current CER: 0.218
```

```
Processing batch 114
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['- private (sichtbar nur innerhalb der Klasse)', 'Deitandteil eines Komplexen von Bedingungen,
der', 'erführreich das Monusal Legitoren physischen Eurogeres für die']
Label texts: ['- private (sichtbar nur innerhalb der Klasse)', 'Betandteil eines Komplexes von Bedingungen, der'
, 'erfolgreich das Monopol legitimen physischen Zwanges für die']
Current CER: 0.17439293598233996
```

```
Processing batch 115
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
```

```
Predicted texts: ['- Nähe: Eil', '- resistent: Lobbung verbiefen', 'Vasser Leutiges Thomas liegt beweit den beo  
blichen Fortschreibt. Wies sich ihr den technischen']  
Label texts: ['- Nähe: Eifel', '- restrict: Löschung verbieten', 'Unser heutiges Thema liegt beim technischen Fo  
rtschritt. Wie seht ihr den technischen']  
Current CER: 0.20440251572327045
```

```
Processing batch 116  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 35])  
Predicted texts: ['=> Umgang mit fehlenden Werten, abhängig von ihr Struktur', 'Bei der lernen Programen wird di  
e ablände Variabt, da die einerkonkonkeiten der', 'hlt schnee Hergeordnete und hat aus einer einer Einerwasserwar  
tung von 20 Jahren.']  
Label texts: ['=> Umgang mit fehlenden Werten, abhängig von ihrer Struktur', 'Bei der linearen Regression wird d  
ie abhängige Variable y als Linearkombination der', 'hat schwere Herzprobleme und hat nur eine Lebenserwartung v  
on 30 Jahren.']  
Current CER: 0.1891891891891892
```

```
Processing batch 117  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 22])  
Predicted texts: ['- spezialisierte Klasse = Unterklasse (Subb class)', '- Effenberungsprochte nach nicht abgesc  
hlossen', 'und Aufseher belehnt werden.']  
Label texts: ['- spezialisierte Klasse = Unterklasse (subclass)', '-> Offenbarungsgeschichte noch nicht abgeschl  
ossen', 'und Aufseher belohnt werden.']  
Current CER: 0.08333333333333333
```

```
Processing batch 118  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 37])  
Predicted texts: ['Wissen garten an die Stadt, Nauer', 'Studenten nutzen die Campus-Firmenstudien.', '- Infrund  
Tiefe zum Testat']  
Label texts: ['Wiesen grenzten an die Stadt, Mauer', 'Studenten nutzen die Campus-Fitnessstudios.', '- Infos und  
Tipps zum Testat']  
Current CER: 0.18429003021148035
```

```
Processing batch 119  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 33])  
Predicted texts: ['- Ein Geschäftsprozess wird durch durch eine Kollektion von Szenarios bekumentiert', 'Nachtei  
le Einlüngesten', 'immer warm']  
Label texts: ['- Ein Geschäftsprozess wird durch eine Kollektion von Szenarios dokumentiert', '- Nachteile Einli  
niensystem', '- immer warm']  
Current CER: 0.15666666666666668
```

```
Processing batch 120  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 47])  
Predicted texts: ['Wasser ist nissgesamt ein viel gebingtes Gott und wird fret innerall', '- Struktur -> Organis  
ationsdesignen (Organigen)', 'außgebaut werden oder Emspene syntdeckiert werhin.']
```



```
Label texts: ['Wasser ist insgesamt ein viel gefragtes Gut und wird fast überall', '- Struktur -> Organisationsdiagramm (Organigramm)', 'aufgebaut werden oder Enzyme synthetisiert werden.']
Current CER: 0.3241590214067278
```

```
Processing batch 121
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Gruppe kleine sind als die Erfahrungen bei einer momenten Synapse.', '- Elektische Bahn ausgebaut, gut, passiert', '-> Selbere Ereignisse der GG gelangen zulten in SP']
Label texts: ['Synapse kleiner sind als die Einfaltungen bei einer normalen Synapse.', '- Elektrische Bahn ausgebaut, gut, preiswert', '-> Seltene Ereignisse der GG gelangen selten in SP']
Current CER: 0.12755102040816327
```

```
Processing batch 122
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['-> es kann nicht direkte mag. Anziehung sein', '- Zulieferantenwahnemenenen Körperien mit Unternehmen', 'Professional Choicee und Spieltheorie']
Label texts: ['-> es kann nicht direkte mag. Anziehung sein', '- Zuliefererunternehmen kooperieren mit Unternehmen', 'Rational Choice und Spieltheorie']
Current CER: 0.18209876543209877
```

```
Processing batch 123
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 42])
Predicted texts: ['Internet ermöglicht Kommunikation.', 'seum Kulturwissenartoßen oder zu Kulterbeziehungen.', 'Schlettel ab, sondern auch in der']
Label texts: ['Internet ermöglicht Kommunikation.', 'zum Kulturzusammenstoß oder zu Kulturbeziehungen.', 'Schlachtfeld ab, sondern auch in der']
Current CER: 0.2729591836734694
```

```
Processing batch 124
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['- Arbeitsenregion', '- Eitungsierung des Autombratels und Deregulierung von von von Vort', 'no']
Label texts: ['- Arbeitsmigration', '- Liberalisierung des Außenhandels und Deregulierung im Inneren', 'ho']
Current CER: 0.2698412698412698
```

```
Processing batch 125
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['-> Gedanken an den Tod, meht sich Sorgen', 'an der Postsynapse -> Dauerhafte Öffnung', 'Sommerkklamend ist eines meiner Lieblingspröle.']
Label texts: ['-> Gedanken an den Tod, macht sich Sorgen', 'an der Postsynapse -> Dauerhafte Öffnung', 'Sonnenblumenöl ist eines meiner Lieblingsöle.']
Current CER: 0.07446808510638298
```

```
Processing batch 126
```

```
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['- Reizent- Steing', '- schnelle Implementationen der Alternativeen', 'Cloud-Speicher ermöglic
ht den Zugriff von überall.']
Label texts: ['Review-Sitzung', '- schnelle Implementationen der Algorithmen', 'Cloud-Speicher ermöglicht den Zu
griff von überall.']
Current CER: 0.16363636363636364
```

```
Processing batch 127
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['- SJF ist meisten schlechter bzt langsamer', '-> wenig bis keine Transporten', 'Die Versöhnun
gsgewend Aufsse 7.11.10']
Label texts: ['- SJF ist meistens schlechter bzw langsamer', '-> wenig bis keine Transportkosten', 'Die Versöhnu
ngszene Analyse']
Current CER: 0.18360655737704917
```

```
Processing batch 128
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['- Internationale Beziehungen', '- große unbestimmte Fläche', 'Gewinnstrukteilt:']
Label texts: ['- Internationale Beziehungen', '- große unbestimmte Fläche', 'Gemeinsamkeit:']
Current CER: 0.2033898305084746
```

```
Processing batch 129
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 44])
Predicted texts: ['Use Case auf charste Erste beift Kontsteilungen Zusammenmmenhang zur "Umswelt"', '- "D" : "De
pendenden - Investien-Prinzip"', 'Wielflichen werden oft durch Überweitung zerstöt, daraus folgt die zerönung']
Label texts: ['"Use Case auf oberste Ebene heißt Kontextdiagramm Zusammenhang zur ""Umwelt""', '- "D": "Depende
ncy-Inversion-Prinzip"', 'Weideflächen werden oft durch Überweidung zerstört, daraus folgt die Zerstörung']
Current CER: 0.2641509433962264
```

```
Processing batch 130
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['keit Zugriff auf Firebird PHP', 'Wandel der Nachk ausprührung', '- Reprakt wird ausgerenfür B
oden, werden von Basstes Probleten vieln']
Label texts: ['Kein Zugriff auf Firebird API', 'Wandel der Wahlkampfführung', '- Republik wird ausgerufen in Bad
en, werden von Bundes Rebellen überrascht']
Current CER: 0.1802030456852792
```

```
Processing batch 131
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['J. Was kurzzeichnacht die Schwarte Romantik?', 'wird Energie von außen benötigt. Ein Motor sch
liept die', '- Mehrsprachigkeit schalzeptiert und nicht verötnten']
```

```
Label texts: ['5. Was kennzeichnet die Schwarze Romantik?', 'wird Energie von außen benötigt. Ein Motor schließt die', '- Mehrsprachigkeit soll akzeptiert und nicht verpöht werden']
Current CER: 0.10597826086956522
```

```
Processing batch 132
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['- Debugruption DNA und nach mehr Einfluss auf andere', 'undere Interrupts', '1.Bare den Entschheidungsgebautraum von das Wurzel hat auf']
Label texts: ['- Behauptung Ö-U und mehr Einfluss auf andere', 'andere Interrupts', '1. Baue den Entscheidungsba um von der Wurzel her auf']
Current CER: 0.21453287197231835
```

```
Processing batch 133
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Wohür ist die rechte Gehirnhälfte zuständig?', 'Forschungsverichtungen ermöglichen Lebstötigkeiten.', 'Gührenenenen einer dara, das sich das lehr Leser ein gemas ziel von Ort der']
Label texts: ['Wofür ist die rechte Gehirnhälfte zuständig?', 'Forschungseinrichtungen ermöglichen Labortätigkeiten.', 'führen ferner dazu, dass sich der Leser ein genaues Bild vom Ort der']
Current CER: 0.1536231884057971
```

```
Processing batch 134
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['- Vermeide Unterstellungen desgebive Fragen', 'der entsprechend verhalten sie sich verändert.', 'mit das er seine Menskripte verbrennt hat. Danach kommt']
Label texts: ['- Vermeide Unterstellungen & suggestive Fragen', 'dementsprechend verhalten sie sich verändert.', 'mit das er seine Manuskripte verbrannt hat. Danach kommt']
Current CER: 0.1235632183908046
```

```
Processing batch 135
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['-10,8°C Durchschnittsunspester', '- Text ist in zwei gegrafische Regionen aufgeteilt', '-> Sc hwandkt:']
Label texts: ['- 18,2 °C Durchschnittstemperatur', '- Text ist in zwei geografische Regionen aufgeteilt', '-> Sc hwierigkeit:']
Current CER: 0.18421052631578946
```

```
Processing batch 136
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['unterschiedliche Größe ihrer (Ober) und können sich', 'Wenn er genau des schildet, was ihren', 'nul-Werte in Potensreiten']
Label texts: ['unterschiedliche Größe ihrer (Ohren) und können sich', 'wenn er genau das schildert, was ihnen', 'null-Werte in Datensätzen']
Current CER: 0.12455516014234876
```

```
Processing batch 137
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['1. Supranische Macht zu Buchüberfall (zahllich Gegrittet)', 'Die Methoden werden nicht immer
explizit aufgeführt.', 'Kulter und Netnens sicht verbotet macht.']
Label texts: ['1. Sporadische Macht z.B. Banküberfall (zeitlich begrenzt)', 'Die Methoden werden nicht immer exp
lizit aufgeführt', 'Kultur und Weltansicht vertraut macht.']
Current CER: 0.14322916666666666
```

```
Processing batch 138
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['Vegreifelt fasst ohne Grund', '- Herrscher, Mensch Nette seine sozwicht aus ohne die Beuchten
von Gesetzen', '- identische Rechte für Arbeiter, Angestellte und Bennte']
Label texts: ['Verzweifelt fast ohne Grund', '- Herrscher, Monarch übt seine Herrschaft aus ohne die Beachtung
von Gesetzen', '- identische Rechte für Arbeiter, Angestellte und Beamte']
Current CER: 0.14423076923076922
```

```
Processing batch 139
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Insgerend bestnmen Reizstoffe und Reizdauer ab und wie stark ein', 'aufmert die ersten Wasser
erschreiben hat wo so so so er z. Jahre akt war.', 'Nachteile Subdringsystemen']
Label texts: ['Insgesamt bestimmen Reizstärke und Reizdauer ob und wie stark ein', 'aufpasst da er seinen Vater
verloren hat wo er 2 Jahre alt war.', 'Nachteile Stabliniensystem']
Current CER: 0.22131147540983606
```

```
Processing batch 140
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- Der Stötte Punkt war die schlechte wirtschaftliche', '- Kunst, historcher große Versache de
r Auforderbeitungen', 'Nr2.')]
Label texts: ['- Der größte Punkt war die schlechte wirtschaftliche', '- Kunst, Literatur große Versuche der Auf
arbeitungen', 'Nr2.')]
Current CER: 0.14565826330532214
```

```
Processing batch 141
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 46])
Predicted texts: ['Selbstreflexion', '- Jede Sährreinig wird durch durch, durch die oder mehrre Bedingungen Befin
gungen, die zu einem spezellen', '- Typen von Variablen']
Label texts: ['Selbstreflexion', '- Jede Szenario wird durch eine oder mehrere Bedingungen definiert, die zu ein
em speziellen', '- Typen von Variablen']
Current CER: 0.20531400966183574
```

```
Processing batch 142
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
```

```
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Teil ist vom Gartenen (Aggregent) existenzenthängig', 'wird betont, dass all die Seilern nach
den den dennen', 'Nachwissen:']
Label texts: ['Teil ist vom Ganzen (Aggregat) existenzabhängig', 'wird betont, dass all die Sachen nach denen',
'Nativismus:']
Current CER: 0.2579505300353357
```

```
Processing batch 143
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['Elektronische Signahrten von einfahren Vertrrten Verträge.', '- Buborns', 'Update - Vektorric
h aus der eingesen Vergangenheit haben höheres Gewicht']
Label texts: ['Elektronische Signaturen vereinfachen Vertragsabschlüsse.', 'Buttons', 'Update - Vektoren aus der
jüngeren Vergangenheit haben höheres Gewicht']
Current CER: 0.18085106382978725
```

```
Processing batch 144
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['- 2. Faktoren zur überprüfen', 'wie ein Himmel so auf - bebotet (wo) versprechen wir, so z
u lebren,', '- Man will nicht anders sein, Seine Störks kweisen']
Label texts: ['- 2 Faktoren zum überprüfen', 'wie im Himmel, so auf bietet (s.o.) versprechen wir, so zu leb
en,', '- Man will nicht anders sein, seine Stärke beweisen']
Current CER: 0.14124293785310735
```

```
Processing batch 145
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Beine Zielsschehrung) - " so "selten" Gruppen häufigen, aber', 'Der Text stein groß unterw
issentschaftlicher Text', 'im OOA Modell Objekte & Attribute identifizieren']
Label texts: ['(keine Zufallsschwankung) -> "seltenere" Gruppen häufiger, aber', 'Der Text ist ein popularwis
sentschaftlicher Text.', 'im OOA Modell Objekte & Attribute identifizieren']
Current CER: 0.15211970074812967
```

```
Processing batch 146
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Unterteilung in zwei Häfen', '- großer Vorgerten', '- aufgabenorientiert']
Label texts: ['Unterteilung in zwei Hälften', '- großer Vorgarten', '- aufgabenorientiert']
Current CER: 0.1
```

```
Processing batch 147
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['gernden Draht.', '- eine grändern ihren eine eigenenen hüten -> Interviewes Leben entstand',
'Selbstwahrnehmung']
Label texts: ['geraden Draht.', '- einige gründeten ihre eigenen Läden => neues Leben entstand', 'Selbstwahrnehm
ung']
Current CER: 0.12813370473537605
```

```
Processing batch 148
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['Datenbanken speichen Informationen.', 'Die Qualle treibt durch das Meer.', 'sonders der Leser
einem großen Interruptieus.']
Label texts: ['Datenbanken speichern Informationen.', 'Die Qualle treibt durch das Meer.', 'sodass der Leser ein
en großen Interpretations -']
Current CER: 0.15254237288135594
```

```
Processing batch 149
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['- Verlust fachlicher Relevanz durch...', 'der Niederkunde. Der Zusammensrechluss dient zur Ve
rwaltung von', 'sicht und die Umweltschieden immer größer werden, ist es wahrscheinlich,']
Label texts: ['- Verlust fachlicher Relevanz durch ...', 'der Niederlande. Der Zusammenschluss dient zur Verwalt
ung von', 'sinkt und die Umweltschäden immer größer werden, ist es wahrscheinlich,']
Current CER: 0.15789473684210525
```

```
Processing batch 150
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['zeust Kreuzprodukt AB aus 18,8, dem Gruppenprodukt 18% aus 1830', 'Mutenzie:', 'Lebensbedingu
ngen']
Label texts: ['zuerst Kreuzprodukt AB aus A & B , dann Kreuzprodukt ABC aus AB & C', 'Mutexe :', 'Lebensbedingun
gen']
Current CER: 0.17049180327868851
```

```
Processing batch 151
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['Studenten diskieren über aktuelle Themen.', 'Datenschutz gewährleistet Privatsphäre.', '- zu
einer Schicht gehören Personen mit ähnbliedder sozialkommiertiger Lage']
Label texts: ['Studenten diskutieren über aktuelle Themen.', 'Datenschutz gewährleistet Privatssphäre.', '- zu e
iner Schicht gehören Personen mit ähnlicher sozioökonomischer Lage']
Current CER: 0.08928571428571429
```

```
Processing batch 152
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['Die Viel singen heute schaut.', 'zum Stromen).', '- abgegrenzte Verantwortungsbereich']
Label texts: ['Die Vögel singen heute sehr laut.', 'zum Strom).', '- abgegrenzte Verantwortungsbereich']
Current CER: 0.09569377990430622
```

```
Processing batch 153
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
```

```
Predicted texts: ['Ein Apfel fällt von Baum.', '-> vereit zusammenwenden', 'Zustandsübergang (Transition)']  
Label texts: ['Ein Apfel fällt vom Baum.', '-> vereint zusammenstehen', 'Zustandsübergang (Transition)']  
Current CER: 0.12681159420289856
```

```
Processing batch 154  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 43])  
Predicted texts: ['- Feinde können zu freunden werden (Frankreich-Deutschland)', 'Quantitative Datenerfassung',  
'Wohnhäuser an Stad']  
Label texts: ['- Feinde können zu Freunden werden (Frankreich - Deutschland)', '- Quantitative Datenerfassung',  
'Wohnhäuser am Stadtrand']  
Current CER: 0.15363128491620112
```

```
Processing batch 155  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 34])  
Predicted texts: ['Im Winter liegt oft Schnee breußen.', '- Einigkeit von nicht Thema oder Rechte und, Freibt',  
'Wir treffen muss am nächten Sonntag.']  
Label texts: ['Im Winter liegt oft Schnee draußen.', '- Einigkeit war nicht Thema sonder Rechte und Freiheit', '  
Wir treffen uns am nächsten Sonntag.']  
Current CER: 0.22631578947368422
```

```
Processing batch 156  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 39])  
Predicted texts: ['- Induktive Suche nach Klassselen Zusammenhänzungen zwischen', 'und dem Rischlag in klar eine  
n. Wir Menschen vielen im Menschen immer aus bis besse für uns selber', '- Hauptressenzeit (bis 5 Tage Ionen Mom  
ent) im Jhr und Armit']  
Label texts: ['- Induktive Suche nach kausalen Zusammenhängen zwischen', 'auch einen Rückschlag in Kauf nehmen.  
Wir Menschen wollen immer nur das beste für uns selber', '- Hauptregenzeit (bis 5 Tage / Monat) im Juli und Augu  
st']  
Current CER: 0.2639175257731959
```

```
Processing batch 157  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 34])  
Predicted texts: ['Ziele von Parteien', 'Attribute mit ihren Werten', 'gibt seine Unterschrift ab']  
Label texts: ['- Ziele von Parteien', 'Attribute mit ihren Werten', 'gibt seine Unterschrift ab']  
Current CER: 0.12968299711815562
```

```
Processing batch 158  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 21])  
Predicted texts: ['- verbunden mit der Natur', 'Hast du meinen Hund gesehen?', '- Test des Systems als Ganzes']  
Label texts: ['- verbunden mit der Natur', 'Hast du meinen Hund gesehen?', '- Test des Systems als Ganzes']  
Current CER: 0.16030534351145037
```

```
Processing batch 159  
Batch keys: dict_keys(['pixel_values', 'labels'])
```

```
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['die eine unternehmensische', 'fragen:', '- vergene DNA Jahren Religion stark an Einfluss in d
ie Staatverschung verberen']
Label texts: ['die eine unternehmerische', 'Folgen:', '- vergangene 200 Jahren Religion stark an Einfluss in die
Staatsordnung verloren']
Current CER: 0.13095238095238096
```

```
Processing batch 160
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 37])
Predicted texts: ['Walt und versichert ihr, dass es ihr gut geht gehe.', 'Zusammenlassend lässt sich spren, dass
Schnelle überzugt an Probeits Nutnung ist', 'und Vorgänge an 8. Oktober 1989 aus der']
Label texts: ['schaft und versichert ihr, dass es ihr gut gehe.', 'Zusammenfassend lässt sich sagen, dass Schnel
le überzeugt von Pinkers Meinung ist', 'und Vorgänge am 9. Oktober 1989 aus der']
Current CER: 0.25415676959619954
```

```
Processing batch 161
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['- Eigenschen guter Messung', 'Unbewusste Verläunkte, Kontlheit', '- Zusammenhängee der Wirklic
h']
Label texts: ['- Eigenschaften guter Messung', 'Unbewusste / erklärliche, Krankheit', '- Zusammenhänge der Wirkl
ichkeit']
Current CER: 0.10749185667752444
```

```
Processing batch 162
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Studententenkonibt', '6) - Strombrose 2 Trennung der Chromorien durch die Grundfassern, ziel
en transportiert', '-> es entsteht ein magen Feld durch durch den Induktionsen']
Label texts: ['Studentenrabatt', '6.) Anaphase 2 Trennung der Chromatiden durch die Spindelfasern, Zellpolen tra
nsportiert', '-> es entsteht ein magn. Feld durch den Induktionsstrom']
Current CER: 0.20789473684210527
```

```
Processing batch 163
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['Jugendlichen so beleitet ist.', 'von lautfändlichen) durchstehmenschungsfeilige Proteinsetzen
enen.', 'Wenn UT-Spezialen aus einem Land auch auswenden, dann und Objeren wird die Böh rung']
Label texts: ['Jugendlichen so beliebt ist.', 'von (ausländischen) durchsetzungsfähige Privatinvestoren.', 'Wenn
IT-Spezialisten aus einem Land auswandern, dann wird die Digitalisierung']
Current CER: 0.30641330166270786
```

```
Processing batch 164
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
Predicted texts: ['- Bedartsprinzip: Sicherung der Grundbedürfnisse', '- dünne Humusschicht', 'Wiss ist Episch f
```



```
ür die Kaiserreit? Gesellschaftliche Stände, Aufstättigkeiten']  
Label texts: ['- Bedarfsprinzip: Sicherung der Grundbedürfnisse', '- dünne Humusschicht', 'Was ist typisch für d  
ie Kaiserzeit? Gesellschaftliche Stände, Auffälligkeiten']  
Current CER: 0.15223097112860892
```

```
Processing batch 165  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 24])  
Predicted texts: ['Gibt es mir öffentliche Toiltten.', '- Mere SP aufgrund der gewonnenenenen Einkommtnisse', '3.  
Nationale Nicht-Regierungsorganisationen) (Muss)']  
Label texts: ['Gibt es hier öffentliche Toiletten.', 'Neue SP aufgrund der gewonnenen Erkenntnisse', '3. Nationa  
le Nicht-Regierungsorganisationen (NGOs)']  
Current CER: 0.15160349854227406
```

```
Processing batch 166  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 39])  
Predicted texts: ['Digitale Kalten der organisieren den Zeitaben.', 'Ein Regentag länst zum lassen ein.', '- Zie  
l: Bestätigung einer Theorie auf ihre Wertigkeit']  
Label texts: ['Digitale Kalender organisieren den Zeitplan.', 'Ein Regentag lädt zum Lesen ein.', 'Ziel: Bestäti  
gung einer Theorie auf ihre Wertigkeit']  
Current CER: 0.2776349614395887
```

```
Processing batch 167  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 22])  
Predicted texts: ['Validiertät: - Durch Material gebrieben', 'Libraries:festivals förren das gewirbare Worte.',  
'gehaffen.']  
Label texts: ['Validität: - Durch Material getrieben', 'Literaturfestivals feiern das geschriebene Wort.', 'gesc  
haffen.']  
Current CER: 0.14893617021276595
```

```
Processing batch 168  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 24])  
Predicted texts: ['-> zugrundelegende Natrie der innerialistischen Großmöchte', 'Die Serech hirht auf dem Teil.'  
, '- Individen / Personen - soziale Kollektive / Organisationen']  
Label texts: ['-=> zugrundeliegende Motive der imperialistischen Großmächte _', 'Die Seerose blüht auf dem Teich.'  
, '- Individuen | Personen \\\ - soziale Kollektive / Organisationen']  
Current CER: 0.16463414634146342
```

```
Processing batch 169  
Batch keys: dict_keys(['pixel_values', 'labels'])  
Batch pixel_values shape: torch.Size([8, 3, 384, 384])  
Batch labels shape: torch.Size([8, 128])  
Generated outputs shape: torch.Size([8, 33])  
Predicted texts: ['- Freiheit: Aufgabe die Freibit der Menschen', 'Online-Bewegen erleichten den Bewertungs.', 'G  
esellschaft bezugruchen.']  
Label texts: ['- Freiheit: Aufgabe die Freiheit der Menschen', 'Online-Bewerbungen erleichtern den Bewerbungspro  
zess.', 'Gesellschaft beanspruchen.']  
Current CER: 0.19012345679012346
```

```
Processing batch 170
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['- Kontrolle', 'gesamten Familie', 'Crowd soncing unterzt kolktive Intelligenz.']
Label texts: ['- Kontrolle', 'gesamten Familie', 'Crowdsourcing nutzt kollektive Intelligenz.']
Current CER: 0.13818181818181818

Processing batch 171
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['Bibliothekablange helfen bei der Bebanke.', 'Asso: Typ den Variablen wird dynamisch entschied en.', 'Kommt du mir sagen wie viel Uler es ist?']
Label texts: ['Bibliothekskataloge helfen bei der Recherche.', 'Also : Typ der Variable wird dynamisch entschied en.', 'Kannst du mir sagen wie viel Uhr es ist?']
Current CER: 0.19498607242339833

Processing batch 172
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['- Eigenschaften der wennstelcheiten Daten aus dem Anwendgrunden mit zuge Operationen Operatio nen Organisationen', '- Kommunissionsismus: Erfolg und Bestand des Systems möt sickeren', '- Clenunge - Theori e']
Label texts: ['- Eigenschaften der wesentlichen Daten aus dem Anwendungsbereich mit zugehörigen Operationen', '- Kommunismus: Erfolg und Bestand des Systems mitsichern', '- Cleavage - Theorie']
Current CER: 0.25925925925925924

Processing batch 173
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['-> soziale Herkunft', 'nicht erwähnt werden. Daraus folgte Verwirung', 'Nachmenge']
Label texts: ['-> soziale Herkunft', 'nicht erwähnt wurden. Daraus folgte Verwirrung', 'Nachfrage']
Current CER: 0.08

Processing batch 174
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['5.) In jeder Tochterzelle bildung eines Spindleitsprozuments', 'Unteilen zu verlindern', '- U nterhaltung erleichtert']
Label texts: ['5.) In jeder Tochterzelle bildung eines Spindelfaserapparats', 'Urteile zu verhindern', '- Untert eilung erleichtert']
Current CER: 0.12280701754385964

Processing batch 175
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['Frücher 600 Eurocher mit Nutz Gützen in einer Landstadt', '1) - Kapräisten / jetzte, Nachvers
```

```
chieden / Gebilde', 'Antitbessen der Gegensatz zwischen gestoren']
Label texts: ['Früher 6000 Einwohner mit Nutz Gärten in einer Landstadt', '1) IT-Spezialisten / Ärzte, Krankensc
hwestern / Geistliche', 'Antithesen der Gegensatz zwischen gestern']
Current CER: 0.23848238482384823
```

```
Processing batch 176
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['- Positionung der Schule mit gewichtig zusammen erschlossen Antworten', '- In Wissensveralls
chaft "wo Wissen im Mittelpunkt', '3. Ausswort, Zusammenhörnung und Versung']
Label texts: ['- Bestimmung der Skala mit gewichteter Zusammensetzung der gegebenen Antworten', '- In "Wissensge
sellschaft", wo Wissen im Mittelpunkt', '3. Abschlusswort, Zusammenfassung und Warnung']
Current CER: 0.23283582089552238
```

```
Processing batch 177
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['- Talente helten dem Unternehmen tehler zu finden', '- Jahresabsabsabsch schluss aufstellen',
'- kann mit einer Aktion verbunden sein']
Label texts: ['- Talente helfen dem Unternehmen Fehler zu finden', '- Jahresabschluss aufstellen', '- kann mit e
iner Aktion verbunden sein']
Current CER: 0.11688311688311688
```

```
Processing batch 178
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['Wenn nicht Natur, dassrauch suchen?', '- wird starkk kritisiert', '- sollte Kürfen durch nich
t ausagezahlzusseit zu werden']
Label texts: ['- Wenn nichts Wahr, warum danach suchen?', '- wird stark kritisiert', '=> Dritte dürfen davon nic
ht ausgeschlossen werden']
Current CER: 0.19858156028368795
```

```
Processing batch 179
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- intertern einzelnde Teilregene die ihren Dad sichtern.', 'Geselle beschrieben, was er auf s
einer Handnung', 'Ein Moment verweilt.']
Label texts: ['- intern: einzelne Teilregime die ihren Bestand sichern', 'Geselle beschrieben, was er auf seiner
Wanderung', 'Ein Moment verweilt.']
Current CER: 0.18786127167630057
```

```
Processing batch 180
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Nr.2) - Buste, Tage, Tahrader', 'Übern der Durchführung von QS:', 'Die Objeker      7.11.10.2
0.2022']
Label texts: ['Nr.2.) - Busse, Züge, Fahrräder', 'Ebenen der Durchführung von QS:', 'Die Physiker      25.10.202
2']
Current CER: 0.21641791044776118
```

```
Processing batch 181
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['- Prozesse der Machtbildung (Repital)', 'KI optimiert Prozesse.', 'sie sich man den Tillern g
ewankten und aus den Klker für']
Label texts: ['Prozesse der Machtbildung (Popitz)', 'KI optimiert Prozesse.', 'sie sich von den Tellern genommen
und aus dem Keller für']
Current CER: 0.1592920353982301
```

```
Processing batch 182
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['- produktive Artantsspaltung innerhalb eines Hobtts', '- Durch Material getrieben', '1. Bedar
fsmenge beliebigteiler']
Label texts: ['- reproduktive Artaufspaltung innerhalb eines Habitats', 'Durch Material getrieben', '1. Bedarfsm
enge beliebig teilbar']
Current CER: 0.11232876712328767
```

```
Processing batch 183
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['-> materielele Einheiten mit bestimmter Bedeutung', 'Einflaltung der eigenen Persönlichkeit',
'- funktionale und nicht-funktionale Anforderungen']
Label texts: ['-> materielle Einheiten mit bestimmter Bedeutung', 'Entfaltung der eigenen Persönlichkeit', '- fu
nktionale und nicht-funktionale Anforderungen']
Current CER: 0.08408408408408409
```

```
Processing batch 184
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Durchlung der Wirklichkeit und müssen auch der Nordblickkeit', 'Kontext genug', 'Zum schnen t
refflichen verwendeter Abbindungen']
Label texts: ['Darstellung der Wirklichkeit und müssen anhand der Wirklichkeit', 'konkret genug', 'Zum schnellen
Auffinden verwendeter Abbildungen']
Current CER: 0.2643312101910828
Skipping batch 185
Skipping batch 186
```

```
Processing batch 187
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Harderare führt aufehle aus.', '- fehlende Stabilität / Schutz', '- Interaktion von Elemente
n']
Label texts: ['Hardware führt Befehle aus.', '- fehlende Stabilität / Schutz', '- Interaktion von Elementen']
Current CER: 0.20065789473684212
```

```
Processing batch 188
Batch keys: dict_keys(['pixel_values', 'labels'])
```

```
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['1..3', '- Rational Choiceal Choice und Spieltheorie', 'Digitale Bildung erweitert den Zug zu Wissen.']
Label texts: ['1..3 .. hat mindestens 1, 2 oder höchstens 3...', '- Rational Choice und Spieltheorie', 'Digitale Bildung erweitert den Zugang zu Wissen.']
Current CER: 0.3193717277486911
```

```
Processing batch 189
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['- Negal liegt rördlich von Inden und sindlich von China', 'Dodiere Firmschlasse biete beitere Tänig von zu eine Humse aus.', '- individeller und flexibleer']
Label texts: ['- Nepal liegt nördlich von Indien und südlich von China', 'Online-Fitnesskurse bieten Training von zu Hause aus.', '- individueller und flexibler']
Current CER: 0.22388059701492538
```

```
Processing batch 190
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['- nur Jachliche Zielsetzung bekommt', '- Umweltschutz (Klimenschutz)', 'mehr als eine Heimat haben']
Label texts: ['- nur fachliche Zielsetzung bekannt', '- Umweltschutz (Klimaschutz)', 'mehr als eine Heimat haben.']
Current CER: 0.08502024291497975
```

```
Processing batch 191
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['- Eine Verängung wird nicht unterstützt', '-> Fragmentärung des Speiches', 'die Marquise die Mutter bietet, eine vielbaume']
Label texts: ['- Eine Vereinigung wird nicht unterstützt', '-> Fragmentierung des Speichers', 'die Marquise die Mutter bittet, eine Hebamme']
Current CER: 0.09578544061302682
```

```
Processing batch 192
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['- Politis: Subrupts der Auswandensantung, so. Akteure => Rölkerges-E', 'die eron nutwenden ge hört hat. Manschen', 'Ein Blatt fällt.']
Label texts: ['- Politics: Schauplatz der Auseinandersetzung pol. Akteure => Policy-Cycle', 'die er von Mitmenschen gehört hat. Menschenn', 'Ein Blatt fällt.']
Current CER: 0.20469798657718122
```

```
Processing batch 193
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 18])
Predicted texts: ['-> berufliche Stellung', 'Unternehmen Finanzinstoffel', 'Handnander werden die gend gesuch
```

```
t.']
Label texts: ['-> berufliche Stellung', 'Unternehmen Finanzmittel', 'Handwerker werden dringend gesucht.']
Current CER: 0.14624505928853754

Processing batch 194
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['der Zahläude steht, unterstivt Schön und', '- Tourismusstelbst Grös', '- Unternehmen haben To
hter unternehmen']
Label texts: ['der 2 Schläuche steht, unterstützt Serbien und', '- Terminvorschlag bis', '- Unternehmen haben To
chterunternehmen']
Current CER: 0.19722222222222222

Processing batch 195
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 18])
Predicted texts: ['der steit der beiben erleitet.', '- Artromder Darstellung', 'Känder da.']
Label texts: ['der Streit der beiden eskaliert.', '- Arten der Darstellung', 'Länder da.']
Current CER: 0.11607142857142858

Processing batch 196
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['und deren Befnisse im Vorlesgrund stehen.', 'beeinflussen sieht immer und immer', '- Elimieru
ng zänstlichen Stäufsfaktoren der unabhängig']
Label texts: ['und deren Befugnisse im Vordergrund stehen.', 'beeinflussen sie vergeht immer und immer', '- Elim
inierung sämtlicher Störfaktoren der unabhängigen']
Current CER: 0.13213213213213212

Processing batch 197
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['-> komplex Schnitte als eigene Geschäftsprozesse spezifizieren (include)', '-> Verlustbetbetb
etleitung', '- Sanktionen zeigen Wirkung (Z.10)']
Label texts: ['-> Komplexe Schritte als eigene Geschäftsprozesse spezifizieren (include)', '- Verlustbeteiligung
', '- Sanktionen zeigen Wirkung (Z.150)']
Current CER: 0.2080536912751678

Processing batch 198
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['- wie sich Grund selbst auswählen kann', 'Nr.2 Bevölkerungsentwicklung', 'auf dem Niveau eine
s Kleinkides.']
Label texts: ['- wo sich jemand selbst auswählen kann', 'Nr. 12 Bevölkerungsentwicklung', 'auf dem Niveau eines
Kleinkindes.']
Current CER: 0.13125

Processing batch 199
Batch keys: dict_keys(['pixel_values', 'labels'])
```

```
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['arbeiten (Z.33-72)', 'ein Produkt oder ein neue kalasse für ein bestandts Produkt.', '4. Mens
chen machen Zähr.']
Label texts: ['arbeiten (Z.73-77)', 'ein Produkt oder ein neues Release für ein bestehendes Produkt', '4. Mensch
en machen Fehler.']
Current CER: 0.14385964912280702
```

```
Processing batch 200
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['bessere Zusungen als dielechte Verfertennten', '1) Zurändert werden die Blutprote und die Wei
zteille eines Beines von Frach', '- Rest im angemessenenenenenrenenenverhältnis']
Label texts: ['bessere Lösungen als direkte Verfahren liefern', '1) Zunächst werden die Blutgefäße und die Weich
teile eines Beines vom Frosch', '- Rest im angemessenen Verhältnis']
Current CER: 0.19287833827893175
```

```
Processing batch 201
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['zäuser hat das Spracht die Form das, Massentriert', 'Keine Auswahlgrundgesamtheit beziehreibb
ar', 'dassstellenllen.']
Label texts: ['zäsur hat das Sonett die Form des Alexandriners.', 'Keine Auswahlgrundgesamtheit beschreibbar',
'darstellen.']
Current CER: 0.22895622895622897
```

```
Processing batch 202
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['2) beschreibe und erlasse den Transport von Glasse von Chromosomenraum in die Blutgeöte.', '-
Antwortägernen geben Signale', 'Produktionsmöglichkeiten']
Label texts: ['2.) Beschreibe und erkläre den Transport von Glukose vom Darminnenraum in die Blutgefäße.', '- An
twortkategorien geben Signale', 'Produktionsmöglichkeiten']
Current CER: 0.16712328767123288
```

```
Processing batch 203
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['Nachteilte mit sich stirck. Hing kommt, dass', 'Ein Gedanke fliegt.', '- höhere Lebenshaltung
skosten']
Label texts: ['Nachteile mit sich. Hinzu kommt, dass', 'Ein Gedanke fliegt.', '- höhere Lebenshaltungskosten']
Current CER: 0.09578544061302682
```

```
Processing batch 204
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['Maschiedes Lernen analysiert Muster.', '- Zentänung und Unterstützung der Bevölkerung', '- La
ste Variablen: Kändern nur inheht gemessen werden.']
```

```
Label texts: ['Maschinelles Lernen analysiert Muster.', '- Zustimmung und Unterstützung der Bevölkerung', '- lat
ente Variablen: können nur indirekt gemessen werden.']
Current CER: 0.22118380062305296

Processing batch 205
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['-> Industrieländern sich geringer bis kein Bevölkerungsstige', '- Entstehung -> wie OHG', 'Da
s Trainieren der BR ist ist ein sehr spal aufe Datnengen insflöttert']
Label texts: ['-> in Industrieländern: sehr geringer bis kein Bevölkerungsanstieg', '- Entstehung -> wie OHG', '
Das Trainieren der SVM ist für sehr große Datenmengen ineffizient']
Current CER: 0.19230769230769232

Processing batch 206
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['Die Wälten auf dem Neu sind sehr hoch.', 'Eine neue Sprache zu lernen ist Teil der', 'Einen)b
estanonenenen der Inte oder Leben des Würfe']
Label texts: ['Die Wellen auf dem Meer sind sehr hoch.', 'Eine neue Sprache zu lernen ist Teil der', 'Eiweißbest
andteile der Haare oder Federn des Wirtes']
Current CER: 0.20279720279720279

Processing batch 207
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Designen und Implementationenierung liegen hohe bekannter', 'Der Hase hoppelt durch den Wald.
', 'organische Tätigkeiten & Informationen zum leiten 2 klein oder Organisation begrägten benöglich Qualit']
Label texts: ['Design und Implementierung liegen nahe beieinander', 'Der Hase hoppelt durch den Wald.', 'organis
atorische Tätigkeiten & Maßnahmen zum Leiten & Lenken einer Organisation bezügliche Qualität']
Current CER: 0.22950819672131148

Processing batch 208
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 41])
Predicted texts: ['- Skälrierbare & Hardware-beschlungste Training', '-> Widerspründe werden offenbart', '- Krite
rien für Staat / Erde eines Testlaufs']
Label texts: ['- Skalierbares & Hardware - beschleunigtes Training', '-> hintergründe werden offenbart', '- Krit
erien für Start / Ende eines Testlaufs']
Current CER: 0.19943019943019943

Processing batch 209
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['Dauerhalt eine Temperatur von 25-30°C.', 'Der Blumenarabau in Konia hat mehre ihr und Nachtei
lle.', 'nicht mehr nur das Schlechsfeld sondern']
Label texts: ['Dauerhaft eine Temperatur von 25-30°C.', 'Der Blumenanbau in Kenia hat mehrere Vor- und Nachteile
.', 'nicht mehr nur das Schlachtfeld sondern']
Current CER: 0.12213740458015267
```



```
Processing batch 210
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
Predicted texts: ['- Streiks und Demonstrationen', '=> Alles besprochen', '- Äbrhin der Physiker']
Label texts: ['- Streiks und Demonstrationen', '=> Alles besprochen', '- Ärztin der Physiker']
Current CER: 0.19498607242339833

Processing batch 211
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 40])
Predicted texts: ['Jebogen der Natur und der Wanderschaft.', 'Lisp ziehmert sich durch Markus aus.', 'Studenten
diskitären über akademische Themen.']
Label texts: ['Ichs zu der Natur und der Wanderschaft.', 'Lisp zeichnet sich durch Makros aus.', 'Studenten disk
utieren über akademische Themen.']
Current CER: 0.20195439739413681

Processing batch 212
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 46])
Predicted texts: ['Das Auto fährt sehr schnell.', 'Nurve 2 zeigt das Ergebnis einer Reizung eines Riesen an, wel
ches nach sich zuach zuch sein Männnten', 'Hentzutge werden Nachkommen nicht mehr verwendet.']
Label texts: ['Das Auto fährt sehr schnell.', 'Kurve 4 zeigt das Ergebnis einer Reizung eines Riesenaxons an, we
lches nach sechs Minuten', 'Heutzutage werden Nebelkammern nicht mehr verwendet.']
Current CER: 0.1366120218579235

Processing batch 213
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 42])
Predicted texts: ['Außerdem gab es auch immer neue Informationen', 'Das Staatenhaus betrad aus 150% 1859 Vollkte
niten der 3 Stadten und wurde die', 'oder genutationen einzelne oder mehre individür']
Label texts: ['Außerdem gab es auch immer neue Informationen', 'Das Staatenhaus bestand aus 168 Volksvertretern
der 38 Staaten und wurde alle', 'oder genmutationen einzelner oder mehrer individueller']
Current CER: 0.21136363636363636

Processing batch 214
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['- Schlechte Wirtschaft -> Flucht in den Westen', 'Bsp. Deitungsverität in einer bioethischen
Dekte', 'Aktionspotential ausgeleistet werden kann']
Label texts: ['- Schlechte Wirtschaft -> Flucht in den Westen', 'Bsp. Deutungs rivalität in einer bioethischen De
batte', 'Aktionspotential ausgelöst werden kann']
Current CER: 0.1069182389937107

Processing batch 215
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 45])
Predicted texts: ['Neute verbinden Geröte.', 'Studenten planen ihre Zukunft sorgfällig.', 'sie direkt an und säl
```

```
l sie lernflikern.'])
Label texts: ['Netzwerke verbinden Geräte.', 'Studenten planen ihre Zukunft sorgfältig.', 'sie direkt an und wil
l sie heimführen.'])
Current CER: 0.25089605734767023
```

```
Processing batch 216
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Ich möchte einen Mafte mit Mülleon.', 'Pflichtenhefft', '-omplexe -> Menge der Unterklassen is
t vollständig']
Label texts: ['Ich möchte einen Kaffee mit Milch.', 'Pflichtenheft', '- complete -> Menge der Unterklassen ist v
ollständig']
Current CER: 0.18694362017804153
```

```
Processing batch 217
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 28])
Predicted texts: ['Er spricht dass, dass die BR immer durch', 'zwischen Staat und Gesellschaft als gestört.', 'S
tudentenökonomie bieten Unterkunft.'])
Label texts: ['Er spricht davon, dass die BRD immernoch durch', 'zwischen Staat und Gesellschaft als gestört.',
'Studentenwohnheime bieten Unterkunft.'])
Current CER: 0.25949367088607594
```

```
Processing batch 218
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['2. Skizzieren Sie den Unternehmenskraft?', 'Durch Vorieren der Parameter auf bob kommen wir be
liebig viel Modellten aufstellen', '- 50 Jahre schlosscht das Las im der Natur']
Label texts: ['2. Skizzieren Sie den Unternehmenskreislauf!', 'Durch Variieren der Parameter a & b können wir be
liebig viele Modellfunktionen aufstellen', '- 50 Jahre Sehnsucht des Ichs in der Natur']
Current CER: 0.220873786407767
```

```
Processing batch 219
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 29])
Predicted texts: ['Somit würden sich die Einkommensanteschieden innerhalb der Gesellschaft', 'Fahrischen entsteh
en am Stad => metstand => metspaschnussige Nissenivier', 'aktive Vormacht in Europa']
Label texts: ['Somit würden sich die Einkommensunterschiede innerhalb der Gesellschaft', 'Fabriken entstanden am
Stadtrand => mehrgeschossige Massenmietshäuser', 'religiöse Vormacht in Europa']
Current CER: 0.15
```

```
Processing batch 220
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 40])
Predicted texts: ['- Zusstünmerung Gontaktors zur Einigung', '- Bührggrundfaltung', 'populationen über mehre Jahr
e. Die Populationwahrke in ein Ein Eizialnissen steht in Abhängigkeit']
Label texts: ['- Zustimmung Gorbatschows zur Einigung', '- Hintergrundstrahlung', 'population über mehrere Jahre
. Die Populationsdichte in rel. Einheiten steht in Abhängigkeit']
Current CER: 0.2375366568914956
```

```
Processing batch 221
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['- Basisimation am Beginn der Langen Welle', 'Konfliktändig ist die Gruppe erst wenn sie für W
irtschaft ausschafft', 'Gesetze entscheiden.']
Label texts: ['- Basisinnovation am Beginn der Langen Welle', '- Konfliktfähig ist eine Gruppe erst wenn sie für
Wirtschaft und Gesellschaft', 'Gesetze entscheiden.']
Current CER: 0.12320916905444126
```

```
Processing batch 222
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Die Zeitung erschünd wirdmethodlich.', '2. Modernwissenschaft:', 'Erstellung einer zwöchst- u
nstrakturten Samndung von Anforderungen an']
Label texts: ['Die Zeitung erscheint wöchentlich.', '2. Medienwissenschaft:', 'Erstellung einer - zunächst - uns
trukturierten Sammlung von Anforderungen an']
Current CER: 0.18042813455657492
```

```
Processing batch 223
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Wenn es uns aller gut gut gut, erkennen wir uns nicht aus, wie zu auferen Phancen', '- Chorwe
iler hat eine Fläche von 1940, Wirtschenisch', 'in die zentralen Regionen transportiert. Aus zu den zerden Regio
nen werden die']
Label texts: ['wenn es uns selber gut geht, kümmern wir uns nicht um andere, wie z.B. um unseren Planeten', '- C
horweiler hat eine Fläche von 1,9 km2, linksrrheinisch', 'in die zentralen Regionen transportiert. Aus den zentr
alen Regionen werden die']
Current CER: 0.19318181818181818
```

```
Processing batch 224
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['ein HTML-Element des vorgegaben Typs ohne Kundsicht', 'Kaffte ist der besste Grund vielle Stu
denten.', 'Bestandteile:']
Label texts: ['ein HTML-Element des vorgegebenen Typs ohne Rücksicht', 'Kaffee ist der beste Freund vieler Stude
nten.', 'Bestandteile:']
Current CER: 0.12380952380952381
```

```
Processing batch 225
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Morgen muss ich zum Zahnarzt in der Innenstadt.', 'Nachbareränder sind Nach Namakten, Botsma,
Sambiz, S. R.8.8 Korge', '- Dürfe und Stähte sind nur über iber Teiligung, flad und Mount Everest Everest']
Label texts: ['Morgen muss ich zum Zahnarzt in der Innenstadt.', 'Nachbarländer sind Namibia, Botswana, Sambia,
D. R. Kongo', '- Dörfer und Städte sind nur über Trekkingweg, Pfad und Mount Everest']
Current CER: 0.1661721068249258
```

```
Processing batch 226
```

```
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['- verfruhre ziel', '- Adides AG', '- Einzugt von Eigennten, untergeentischen, Konsorger']
Label texts: ['- verfügbare Zeit', '- Adidas AG', '- Einzug von Eigentum, Wertgegenständen, Konsumgüter']
Current CER: 0.28708133971291866

Processing batch 227
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 39])
Predicted texts: ['- Bischer verwenden vor im Gradientenfahren von "Durchlau) (alle) alle', '- Jedes Land baute
Militativ aus', 'Stichprobe entnomenen und ausgesetzt']
Label texts: ['- Bisher verwenden wir im Gradientenverfahren pro Durchlauf (Epoche) alle', '- Jedes Land baute M
ilitär aus', 'Stichproben entnommen und ausgewertet.']
Current CER: 0.1940700808625337

Processing batch 228
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 37])
Predicted texts: ['-> Beschaffungsprogramm?', 'Die Gänse fliegen in Formation.', 'Orten an Artenstischen Oren']
Label texts: ['-> Beschaffungsproram?', 'Die Gänse fliegen in Formation.', 'Osten am Atlantischen Ozean']
Current CER: 0.22764227642276422

Processing batch 229
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 45])
Predicted texts: ['Ursieht Ausfall mit einemen bei Grunden', 'Beindnungskeiten des Varianz. Dadurch isoliert sic
h die Grunder als Protein (Z-G-D-Gformen) werden)', 'Nr.6']
Label texts: ['Er spielt Fussball mit seinen drei Freunden', 'Bindungsstellen des Carriers. Dadurch ändert sich
die Struktur des Proteins (= Konformationsänderung), wodurch', 'Nr. 6']
Current CER: 0.2810945273631841

Processing batch 230
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['1. Ein Akteur hat Interessen und unfolgt Ziele', '- Vorbeitung auf die Zukunft / Integration'
, '- Eine einzelne Bedingung ist ein nichtin michtinchreiber']
Label texts: ['1. Ein Akteur hat Interessen und verfolgt Ziele', '- Vorbereitung auf die Zukunft / Integration',
'- Eine einzelne Bedingung ist ein nicht hinreichender']
Current CER: 0.11209439528023599

Processing batch 231
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['Mögliche Probleme bei "Antwortäußerungen" :', 'Die schiede Qualle behandelt die Einbrüche', '
Gruppen oder Staaten in den bestehende Nachbütte']
Label texts: ['Mögliche Probleme bei "Antwortäußerungen":', 'Die vorliegende Quelle behandelt die Eindrücke', 'G
ruppen oder Staaten in dem bestehende Konflikte']
```

Current CER: 0.08238636363636363

Processing batch 232

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 28])

Predicted texts: ['Vorberzeitung und zur Ausstellung gesund gesellschaften', '- umfasst alle Maßnahmen für den Verkauf von Produkten', '- Verwüfung und Trockenheit vieler Rogenen, Schrema, Durch Gein']

Label texts: ['Vorbereitung und zur Herstellung gesamt gesellschaftlich', '- umfasst alle Maßnahmen für den Verkauf von Produkten', '- Verwüstung und Trockenheit vieler Regionen, Sahelzone, Durch China']

Current CER: 0.15339233038348082

Processing batch 233

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 30])

Predicted texts: ['amer Wiese soll nochmal Zestromen, dass es nicht um die Hölker der', '- Europäer haben auch andere Schnee', '- Unterschulden: Matrix']

Label texts: ['armen Witwe soll nochmal betonen, dass es nicht um die Höhe der', '- Europäer haben auch andere Schneewörter', 'Staatsschulden : Metrisch']

Current CER: 0.22105263157894736

Processing batch 234

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 24])

Predicted texts: ['Vorlesungen vermitteln Wissen.', 'Heute morgen sich ich sich eine Stereschmappe.', 'von Herkunft oder Zahlgang, kannnt diese / diese']

Label texts: ['Vorlesungen vermitteln Wissen.', 'Heute morgen sah ich eine Sternschnuppe.', 'von Herkunft oder Intelligenz, lernt diese Strukturen']

Current CER: 0.1951219512195122

Processing batch 235

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 33])

Predicted texts: ['die einen einen besondern Fokus auf Metaphern.', '=> Korruptionenenen und jurkte Studente Schebt gegen die Entwicklung als Landes', 'Rechnerischer Wert des Bezugsrechts:']

Label texts: ['dabei einen besonderen Fokus auf Metaphern.', '=> Korruption und instabile Sicherheitslage steht gegen die Entwicklung des Landes', 'Rechnerischer Wert des Bezugsrechts:']

Current CER: 0.17088607594936708

Processing batch 236

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 30])

Predicted texts: ['Sprachen unterscheiden sich in vielen Punkten, nicht oder', 'Forschungsprojekte möglichkeiten Einblicke.', 'Die Ameise trägt eine große Last.']

Label texts: ['Sprachen unterscheiden sich in vielen Punkten, nicht aber', 'Forschungsprojekte ermöglichen Einblicke.', 'Die Ameise trägt eine große Last.']

Current CER: 0.15404040404040403

Processing batch 237

Batch keys: dict_keys(['pixel_values', 'labels'])

```
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['- Wie werden die Daten ausgewartet?', '- Frage zur Prüfungsstrom von mir', '- Kein "Guter" Region']
Label texts: ['- Wie werden die Daten ausgewertet?', '- Frage zur Prüfungsform von mir', '- kein "fester" Beginn']
Current CER: 0.1
```

```
Processing batch 238
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 34])
Predicted texts: ['Definition Massenmedien', '-> Markt- Einteilungsmöglichkeiten', '- Organisationsarbeität']
Label texts: ['Definition Massenmedien', '-> Markt-Einteilungsmöglichkeiten', '- Organisationsautorität']
Current CER: 0.06840390879478828
```

```
Processing batch 239
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 20])
Predicted texts: ['und Einstellung neuer Mitarbeiter', 'Politik werden Partzelt', 'Das Orne "Der gute Mensch von System"']
Label texts: ['und Einstellung neuer Mitarbeiter', 'Politik nach Potzelt', 'Das Drama "Der gute Mensche von Sezu an"']
Current CER: 0.11068702290076336
```

```
Processing batch 240
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['- Eigenschaften guter Messungen', '- 13, 7°95 Ost, 51,1° Nord', 'Drombran von Autokratien']
Label texts: ['Eigenschaften guter Messungen', '- 13,7° Ost ; 31,1° Nord', 'Demokratien und Autokratien']
Current CER: 0.14624505928853754
```

```
Processing batch 241
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 21])
Predicted texts: ['gebötet.', 'Ein Kohtus ist stadig.', '- Kollektiv (Gruppe, Organisationenpotentiale Einheit)']
Label texts: ['getötet.', 'Ein Kaktus ist stachelig.', '- Kollektiv (Gruppe, Organisation, territoriale Einheit)']
Current CER: 0.1652892561983471
```

```
Processing batch 242
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['geigne Möchung aus Text, Tabellen, Grafüllen, Diagrammen', 'Glauen zuene meht.', 'Hilproz']
Label texts: ['geeignete Mischung aus Text, Tabellen, Grafiken, Diagrammen', 'blauen Blume macht.', '14.11.2022']
Current CER: 0.16600790513833993
```

```
Processing batch 243
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['- Anturwandlung durch Mutation, Relkombinationen und Selektion', '- betreibt sich auf auf ein
e Diagnische von, obelt sollt besonders objizt', '- Datenlange']
Label texts: ['- Artumwandlung durch Mutation, Rekombination und Selektion', '- betreibt sich auf eine Pilgerre
ise vor, obwohl sie nicht besonders religiös ist', '- Datenlage']
Current CER: 0.23142857142857143
```

```
Processing batch 244
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['- Leitfaden', 'Im Krankenhaus werden Blünden behandelt.', 'Machtquellen: Solidarität und Orga
nisation']
Label texts: ['- Leitfaden', 'Im Krankenhaus werden Patienten behandelt.', 'Machtquellen: Solidarität und Organi
sierung']
Current CER: 0.08074534161490683
```

```
Processing batch 245
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 35])
Predicted texts: ['Zwei-Chromat- Chromen- entschen sich auch an der Änderblichen au. Metgebose?', 'Eigene Theori
en hinterfragen + prüfen', 'soziale und wirtschaftliche Folgen dieser Entwicklung']
Label texts: ['Zwei-Chromatid-Chromosomen ordnen sich an der Äquatoriallebene an. Metaphase 2', 'Eigene Theorien
hinterfragen + prüfen', 'soziale und wirtschaftliche Folgen dieser Entwicklung']
Current CER: 0.12727272727272726
```

```
Processing batch 246
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 22])
Predicted texts: ['1) Sozialhölfe', '4. Konstruktivismus:', '- Frequenten (Unterschiedungs diffunden Form)']
Label texts: ['11) Sozialhilfe', '4. Konstruktivismus', '- Fragearten (Unterscheidungsdimension Form)']
Current CER: 0.16923076923076924
```

```
Processing batch 247
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 23])
Predicted texts: ['-> wenig extreme Merkmale', 'von der Wiese Julietta die während eines', 'Abschumpsprüfungen t
esten viel Wissen.']
Label texts: ['-> wenig extreme Merkmale', 'von der Witwe Julietta die während eines', 'Abschlussprüfungen teste
n viel Wissen.']
Current CER: 0.14193548387096774
```

```
Processing batch 248
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 33])
Predicted texts: ['Görter sind diese typischen Güter. Sie', '- In den Jahren von 1000 bis 2000 ist der Preis für
```

```
Proböl', 'eine oder mehrere möngeen aushande kommen.'])
Label texts: ['Götter sind keine typischen Götter. Sie', '- In den Jahren von 1990 bis 2010 ist der Preis für Ro
höl', 'eine oder mehrere Wirkungen zustande kommen.'])
Current CER: 0.2028985507246377
```

```
Processing batch 249
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 41])
Predicted texts: ['Forschungsstudition', 'Konig Heinrich bekommen von Papst zum Kirzboben und sollte von seinem
Gott zuändern.', 'gute Konfnehrungszahlungsmötkten']
Label texts: ['Forschungstradition', 'König Heinrich bekam vom Papst einen Kirchenbann und sollte von seinem Amt
zurücktreten.', 'gute Kontrollmöglichkeiten']
Current CER: 0.2708333333333333
```

```
Processing batch 250
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 32])
Predicted texts: ['- einen Viede Rahmenmalungen, keinen Anwendung welcher sich', 'Alle Merkmalsausprägungen von
abhalten Variablen müssen abgeleht sein', 'Markt']
Label texts: ['- ohne visuelle Rückmeldung, keine Ahnung welcher Link', '- Alle Merkmalsausprägungen von abh.|un
abh. Variablen müssen abgedeckt sein', 'Markt']
Current CER: 0.23415977961432508
```

```
Processing batch 251
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ['- gewährleistet Schutz im Alltag', 'Quotenauswahl: nicht zufällige Stichprobe', 'Ich möchte e
inen Katfe mit Mülle.']
Label texts: ['- gewährleistet Schutz im Alltag', 'Qutenauswahl: - nicht zufällige Stichprobe', 'Ich möchte eine
n Kaffee mit Milch.']
Current CER: 0.1751412429378531
```

```
Processing batch 252
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Relationen', 'Der Seerose blüht auf dem Täh.', '- soziales Klima']
Label texts: ['Relationen', 'Die Seerose blüht auf dem Teich.', '- soziales Klima']
Current CER: 0.18209876543209877
```

```
Processing batch 253
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 42])
Predicted texts: ['Java ist plattformunabhängig.', 'Anforderungsaufnahme', 'L4, wird CAM-Pflanzen haben transgüb
er geschlosse Spaltenungen']
Label texts: ['Java ist plattformunabhängig.', 'Anforderungsaufnahme', 'C4- und CAM-Pflanzen haben tagsüber gesc
hlossene Spaltöffnungen']
Current CER: 0.20382165605095542
```

```
Processing batch 254
```



```
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 15])
Predicted texts: ["- fester Bestandteil des BSP'", 'Nr.4.)', 'Vergleich Ethikk zwischen B und GG']
Label texts: ["- fester Bestandteil des BIP's", 'Nr4.)', '- Vergleich Ethik zwischen B und GD']
Current CER: 0.16080402010050251
```

```
Processing batch 255
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Sald die Kartoffstrakzeiten sieht und dadurch die Rauperfliegenprokten', 'Campus Cafis bieten Orte zum Gruppen.', '- Kohle müsste in Australien günstiger produziert werden als']
Label texts: ['Sobald die Kartoffelkäferpopulation sinkt und dadurch die Raupenfliegenpopulation', 'Campus-Cafés bieten Orte zum Entspannen.', '- Kohle müsste in Australien günstiger produziert werden als']
Current CER: 0.1643835616438356
```

```
Processing batch 256
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['4. Beschaffungsgebenditionen aushandeln', 'Die Tage werden immer länger.', 'Er ist einer der ersten Seefahrer die nach']
Label texts: ['4. Beschaffungskonditionen aushandeln', 'Die Tage werden immer länger.', 'Er ist einer der ersten Seefahrer die nach']
Current CER: 0.12941176470588237
```

```
Processing batch 257
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 37])
Predicted texts: ['weibliche Kadieren. Insgesamt bestet besteht das Gedicht aus einer Stadte mit 19 Versen', '- Veränderung der Stärke des Magnetfeldes', '3. Interaktionseffekte zwischen zwei oder mehreren oder']
Label texts: ['weibliche Kadenzen. Insgesamt besteht das Gedicht aus einer Strophe mit 18 Versen', '- Veränderung der Stärke des Magnetfeldes', '8. Interaktionseffekte zwischen zwei oder mehreren der']
Current CER: 0.18099547511312217
```

```
Processing batch 258
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 36])
Predicted texts: ["- schnellerere Verländerung", '- Update', 'Ein Traum erwacht.']
Label texts: ["- schnellere Verladung", '- update', 'Ein Traum erwacht.']
Current CER: 0.09727626459143969
```

```
Processing batch 259
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 24])
Predicted texts: ['2. Internationale Organisationen) (ING', 'Berechnung der Laufzeit:', 'bei der BR beziehen sich stark auf die']
Label texts: ['2. Internationale Organisationen (IGOs)', 'Berechnung der Laufzeit:', 'bei der BRD beziehen sich stark auf die']
```

Current CER: 0.10882352941176471

Processing batch 260

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 24])

Predicted texts: ['- Sprache entsteht durch durch Interaktion zwischen Menschen', 'Die Hochzeit ist sowohl den a
uch der besellschaft', '- Automobilindustie ist bereits in der Rötphase']

Label texts: ['- Sprache entsteht durch Interaktion zwischen Menschen', 'Die Hochzeit ist sowohl dem Druck der G
esellschaft,', '- Automobilindustrie ist bereits in der Reifephase']

Current CER: 0.07417582417582418

Processing batch 261

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 30])

Predicted texts: ['- Ab 20000 sank die Bevölkerung', 'a) Die sicher sind die eine zutreffende Aussage zu lieren'
, 'Chadsformen der Phanze) zudem zuorden zufröllt Markt']

Label texts: ['- Ab 2000 sank die Bevölkerung', 'd.) Wie sicher sind Sie eine zutreffende Aussage zu diesem',
'(Wachstum der Pflanze). -> zudem zerfällt Malat']

Current CER: 0.2276657060518732

Processing batch 262

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 30])

Predicted texts: ['- regulierte gerne - steuerbare gend', 'Richtionsberhandlungungen problemung politische Akt
euren', 'steigt Chromerögert, sinkt die Raupenflänpopulationen und einer Zeit wieder,']

Label texts: ['- regulierte gene - steuer bare gene', 'Positionsbestimmung politischer Akteuren', 'steigt (Phase
nverzögert), sinkt die Raupenfliegenpopulation nach einer Zeit wieder,']

Current CER: 0.2743362831858407

Processing batch 263

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 20])

Predicted texts: ['überprüfung des Jahrgsabschluss.', 'kümmert und sehr denlker ist, dass der', 'empirische Rel
ativ']

Label texts: ['Überprüfung des Jahresabschluss', 'kümmert und sehr dankbar ist, dass er', 'empirische Relativ']

Current CER: 0.07476635514018691

Processing batch 264

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

Batch labels shape: torch.Size([8, 128])

Generated outputs shape: torch.Size([8, 38])

Predicted texts: ['in der Wartnachlange des Semaphlos befördern', 'Wie geht es dir heute?', 'Schrele Merktenpass
ung']

Label texts: ['in der Warteschlange des Semaphors befinden', 'Wie geht es dir heute?', 'schnelle Marktanpassun
g']

Current CER: 0.19710144927536233

Processing batch 265

Batch keys: dict_keys(['pixel_values', 'labels'])

Batch pixel_values shape: torch.Size([8, 3, 384, 384])

```
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 22])
Predicted texts: ['Auflösung der AG', 'Der Machwort grächt sich durch die Erde.', 'Geschäftsprozesses und Szenario']
Label texts: ['Auflösung der AG', 'Der Maulwurf gräbt sich durch die Erde.', 'Geschäftsprozess und Szenario']
Current CER: 0.0970464135021097
```

```
Processing batch 266
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 26])
Predicted texts: ['- alle Sprachen haben gemeinsame Grammatikische Prinzipien', '- Folern sind jedoch bleibt bei den Weiben', 'Warum sind by das erschaststätigen arbeiter?']
Label texts: ['- alle Sprachen haben gemeinsame grammatikalische Prinzipien', '- Federn sind jedoch beliebt bei den Weibchen', 'Warum sind 6% der erwerbstätigen arbeitslos?']
Current CER: 0.13333333333333333
```

```
Processing batch 267
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 31])
Predicted texts: ['Attnmatisch: Schulung eine Computer führung', '- besteht aus 27 Regierungsrechs', 'KI sindier t musschützte Inteilßerz.']
Label texts: ['Automatische Schulung eines Computerdiktions', '- besteht aus 27 Regierungschefs', 'KI simuliert menschliche Intelligenz.']
Current CER: 0.18227848101265823
```

```
Processing batch 268
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 30])
Predicted texts: ['Digitale Landkarten erleichten die Navigation.', '- mehrererereine Gruppels mit unterschiedlich er Menge an Grennen', 'Ausforas bezieht die Anbleme von Arbeitorfund indem es den Einfluss der']
Label texts: ['Digitale Landkarten erleichtern die Navigation.', '- mehrere Genpools mit unterschiedlicher Menge an Genen', 'RMSProp behebt die Probleme von AdaGrad indem es den Einfluss der']
Current CER: 0.2169576059850374
```

```
Processing batch 269
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 27])
Predicted texts: ['Selbstverwirklichung', 'in Zukunft immer über werden.', '2.) Nunwirtschnaptheit -> Ziele Industrie muss']
Label texts: ['Selbstverwirklichung', 'in Zukunft immer älter werden.', '2.) Munitionsknaptheit-> Zivile Industrie muss']
Current CER: 0.17028985507246377
```

```
Processing batch 270
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 25])
Predicted texts: ['Die kann aus dem schönenenenen Beginten durch,', 'Ursachenfaktors konsequierenenen.', '- quant / Qual im 19 / - Ausatz']
Label texts: ['Sie kam aus dem schönen Bergischen Land', 'Ursachenfaktors korrespondieren.', '- quant / qual im
```

```
MM-Ansatz']
Current CER: 0.25316455696202533

Processing batch 271
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([8, 3, 384, 384])
Batch labels shape: torch.Size([8, 128])
Generated outputs shape: torch.Size([8, 38])
Predicted texts: ['Standard-der - Technik bei Adaptien Optimieren', 'Mögische Konstanten', '- entscheidet mit Ra
t über Großteil der Gesetze']
Label texts: ['Stand - der - Technik bei Adaptiven Optimieren', 'Magische Konstanten', '- entscheidet mit Rat üb
er Großteil der Gesetze']
Current CER: 0.11331444759206799

Processing batch 272
Batch keys: dict_keys(['pixel_values', 'labels'])
Batch pixel_values shape: torch.Size([1, 3, 384, 384])
Batch labels shape: torch.Size([1, 128])
Generated outputs shape: torch.Size([1, 20])
Predicted texts: ['- Schuldung bei manwelter Kontlernung:']
Label texts: ['Schulung bei manueller Kodierung:']
Current CER: 0.2727272727272727
Validation CER after evaluation: 0.1789
```

In []:

Testing Code

```
In [ ]: import torch

# Set the model to evaluation mode
model.eval()

# Function to test a single input line
def test_model_on_line(input_text, processor, model, device):
    try:
        # Preprocess the input text
        inputs = processor(text=input_text, return_tensors="pt", padding=True, truncation=True)
        inputs = {k: v.to(device) for k, v in inputs.items()} # Move to device

        # Generate predictions
        with torch.no_grad():
            outputs = model.generate(pixel_values=inputs["pixel_values"])

        # Decode the predicted text
        predicted_text = processor.batch_decode(outputs, skip_special_tokens=True)[0]

        print("\n==== Model Prediction =====")
        print(f"Input Text: {input_text}")
        print(f"Predicted Text: {predicted_text}\n")

        return predicted_text
    except Exception as e:
        print(f"Error during testing: {e}")
        return None

# Example usage:
input_text = "Your sample input line here" # Replace with the actual input
predicted_output = test_model_on_line(input_text, processor, model, device)
```

In []:

Extracting Text From Image

```
In [ ]: import easyocr
reader = easyocr.Reader(['de'])
translator = Translator()
```

```
In [ ]: import PIL
from PIL import ImageDraw
im = PIL.Image.open("/content/Screenshot 2025-01-07 215746.png")
```

```
In [ ]: bounds = reader.readtext('/content/Screenshot 2025-01-07 215746.png', add_margin=0.55, width_ths=0.7, link_thres=0.5)
bounds
```

```
In [ ]: def draw_boxes(image, bounds, color='yellow', width=2):
    draw = ImageDraw.Draw(image)
    for bound in bounds:
        p0, p1, p2, p3 = bound[0]
        draw.line([*p0, *p1, *p2, *p3, *p0], fill=color, width=width)
    return image

draw_boxes(im, bounds)
```

```
In [ ]: text_list = reader.readtext('/content/Screenshot 2025-01-07 215746.png', add_margin=0.55, width_ths=0.7, link_ths=0.7)
        text_list
```

```
In [ ]: text_comb=' '.join(text_list)
        text_comb
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```