Scribosermo: Fast Speech-to-Text models for German and other Languages

Daniel Bermuth, Alexander Poeppel, Wolfgang Reif

University of Augsburg, Institute for Software & Systems Engineering

{daniel.bermuth,alexander.poeppel,reif}@informatik.uni-augsburg.de

Abstract

Recent Speech-to-Text models often require a large amount of hardware resources and are mostly trained in English. This paper presents Speech-to-Text models for German, as well as for Spanish and French with special features: (a) They are small and run in real-time on microcontrollers like a RaspberryPi. (b) Using a pretrained English model, they can be trained on consumer-grade hardware with a relatively small dataset. (c) The models are competitive with other solutions and outperform them in German. In this respect, the models combine advantages of other approaches, which only include a subset of the presented features. Furthermore, the paper provides a new library for handling datasets, which is focused on easy extension with additional datasets and shows an optimized way for transfer-learning new languages using a pretrained model from another language with a similar alphabet.

Index Terms: fast speech to text, multilingual transfer-learning, automatic speech recognition, embedded hardware

1. Introduction

Speech-to-Text models based on neural networks are mostly trained in English and often require large amounts of training resources. But there exist many other languages and those who are interested in training a speech-to-text system for their own language do not always have access to high-performance server hardware. A few papers and projects focus on the aforementioned problems, but most are solving them only partially.

The authors of *IMS-Speech* [1] trained a German STT model, which so far had the best results on the German *Tuda* dataset [2]. In a comparison with Google's STT service (executed 01/2019), their network could outperform it in English as well as in German.

In *VoxPopuli* [3], an approach for training multilingual models using a large unlabeled dataset is investigated. A mix of 50k hours of unlabeled data in different languages from the European Parliament and a comparatively small labeled dataset for semi-supervised training are used. This approach proved very effective and achieves a Word-Error-Rate (WER) of $7.8\,\%$ / $9.6\,\%$ / $10.0\,\%$ in German / Spanish / French on the *CommonVoice* datasets [4], which so far have been the best results on these datasets.

Luo et al. [5] used the same network architecture as this work, but in Nvidia's original implementation, and also trained it for other languages like German or Spanish, using very small datasets and following a different transfer-learning approach of reinitializing the last network layer if the alphabet changes.

Mozilla's *DeepSpeech* project [6] provides pretrained English models that are relatively small and one of the few that are able to run in real-time on a RaspberryPi. It achieves a WER of $7.1\,\%$ on the *LibriSpeech* [7] testset. Some early experiments on multilingual trainings have been run with this network, but performance was much lower than the results presented in the following chapters. They still can be found in the project's repository which is linked later.

Park et al. [8] built a model for embedded devices, which reached a WER of $9.0\,\%$ on *LibriSpeech* and could run on an ARM-Cortex-A57. *Zhang et al.* [9] trained a very small model on a large in-house Chinese dataset which can run faster than real-time on an ARMv7 chip. *He et al.* [10] did train an English model on a very large in-house dataset which can run twice as fast than real-time on a Google-Pixel smartphone.

Ghoshal et al. [11] and *Thomas et al.* [12] did run early explorations of transfer-learning for different languages using deep neural networks. The first approach replaces the last language specific layer of a network with a new one and finetunes the whole network on the new language, while the second uses a multilingual training of the first network layers, and different output layers for each language.

This paper presents a small Speech-to-Text model for German, as well as for Spanish and French, that combines the advantages of the aforementioned approaches. The main contributions of project *Scribosermo* are: (a) The models are competitive with the models from *IMS-Speech* and *VoxPopuli*. (b) Providing pretrained models in multiple languages that can run in real-time even on single-board computers like a RaspberryPi. (c) The models can be trained on a relatively small dataset, like the models from *VoxPopuli* and only require consumer-grade hardware for training. (d) Shows a fast transfer-learning approach with a single step through the concept of alphabet adaption. (e) Improved SOTA performance for German STT.

Furthermore, the paper provides a new library for handling datasets, which is focused on easy extension with additional datasets and shows a simple way for transfer-learning new languages using a pretrained model from a language with an almost similar alphabet, which is demonstrated for English to Spanish and Spanish to Italian transfer-learning.

The training code and models are provided as open source at: *https://gitlab.com/Jaco-Assistant/Scribosermo*

For reasons of readability, the results of the experiments are not presented in full detail, but can instead be found in the project's repository.

2. Pre-processing

The datasets are converted into single channel $16\,\mathrm{kHz}$ audio with *wav* encoding and a *tab* separated *csv* file for each partition, with at least the keys *duration*, *filepath* and *text*. Afterwards an additional data cleaning step is executed. All numbers are converted to their textual form, as well as commonly used units like *kg* or m^2. After replacing some special characters (like *ä*→*ae*), all remaining characters, which do not match the used language's alphabet are removed. All of those rules are collected in a simple *json* file, to ease adding new languages.

In early training executions the transcriptions of some files did not match the recordings, which resulted in errors if they were much too short or much too long. Therefore, and in order to improve training speed, an automatic cleaning process was implemented, which excludes all files matching one of the following metrics:

(1) Audio shorter than half a second.
(2) Audio longer than 30 seconds.
(3) Transcription has more than 512 characters.
(4) Recording is spoken 2x faster than the average.
(5) Less than one character per three seconds is spoken.
(6) (chars/second $<$ average$\times$3) and (duration $>$ average/5).

The second and third items are used to exclude long files to allow for a greater training batch size. The forth and fifth metrics exclude too quickly or too slowly spoken utterances. The last is intended for slow recordings, too, but with an exception for short clips, because those may have longer pauses at the start or end of the recording.

3. Language model

To improve the predicted transcriptions of the trained network, the predictions are rescored with a 5-gram language model. For German a large 8-million sentence collection from [13] is used and combined with the transcriptions from the training dataset. The same text normalization steps as described in the last chapter are executed. In Spanish and French the *Europarl* and *News* sentence collections from [14] are used additionally, and the Italian training sentences are extended with the *Mitads* dataset [15] The language model is created with *PocoLM* [16] and optimized with tools provided by Mozilla's *DeepSpeech* project [6]. For decoding their *ds-ctcdecoder* is used as well. The language models were filtered to a maximum size of 165M n-grams, which results in a size of about 850MB.

4. Experiments with QuartzNet

For the experiments the *QuartzNet* architecture [17] was implemented, using the open source code from Nvidia's *NeMo* project [18] as reference. The QuartzNet architecture (Figure 1) was chosen, because its size is comparatively small, which results in fast inference on standard computers and low power devices. Nvidia provides pretrained weights for English, which reach a greedy WER of 3.8 % on the LibriSpeech devset.

4.1. Reimplementation in TensorFlow

Instead of directly using Nvidia's PyTorch implementation, the network was reimplemented for TensorFlow. The main reason was that the trained network can be directly converted into the *TensorFlow-Lite* format, which greatly improves inference speed on low power devices. Another benefit was that the tools already implemented for Mozilla's *DeepSpeech* framework and some of their data augmentation features could be integrated more easily into the new project.

The pretrained weights from Nvidia's *NeMo* project were transferred layer by layer from the PyTorch format to TensorFlow using *Open Neural Network Exchange (ONNX)* as intermediate format. While this did work well for the network itself, there were problems due to differences in PyTorch's and TensorFlow's spectrogram calculation. To reduce the impact of these, the transferred network was trained for four additional epochs

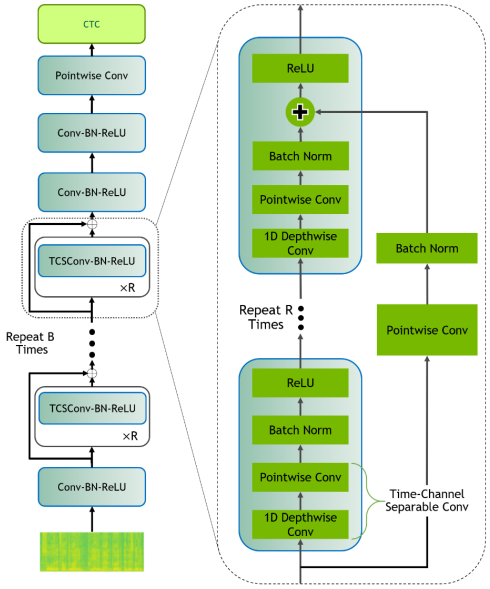

Figure 1: *Network architecture of Nvidia's QuartzNet [17].*

on the LibriSpeech dataset. The performance is still slightly worse than Nvidia's reference implementation, but much better than Mozilla's current *DeepSpeech* release (Table 1).

Table 1: *Performance on English LibriSpeech dataset. Sometimes predictions are rescored with an additional 5-gram language model (LM), else the greedy WER is measured.*

Network	Notes	WER
DeepSpeech	LS-test-clean + LM	7.06 %
QuartzNet15x5	Nvidia, LS-dev-clean	3.79 %
QuartzNet15x5	Converted, LS-dev-clean	5.15 %
QuartzNet15x5	Trained, LS-dev-clean	4.35 %
QuartzNet15x5	above, LS-test-clean	4.57 %
QuartzNet15x5	above, LS-test-clean + LM	3.71 %

4.2. Training in German

For the following trainings the *CommonVoice* (v6) dataset was used. The full training partitions are larger than the amount used in *VoxPopuli* [3], therefore a random subset was selected to match the overall duration. In order to get the same alphabet as in English, the German umlauts ($\ddot{a}, \ddot{o}, \ddot{u}$) have been replaced with their transliteration (ae, oe, ue).

Table 2 shows that the implemented *QuartzNet15x5* network and training procedure, named *Scribosermo* in the table, can outperform other approaches for German speech recognition. The training setup of simply training over a pretrained English network, without any further changes, is straightforward, and does not require a semi-supervised pretraining on multiple languages.

In Table 3 some different training modalities are investigated. The first section uses the complete German training partition of *CommonVoice*, which slightly improves the results. This training took 3 days on a PC with two 1080Ti GPUs, which shows that the training process itself is very fast, too, and can be executed on consumer-grade hardware. The second part shows

Table 2: *German training results. Above networks have been tested on CommonVoice, below on Tuda dataset.*

	Notes	Duration	WER
Scribosermo	using CV v6	314h	7.7 %
VoxPopuli [3]	using CV v5	314h	7.8 %
Luo et al. [5]	greedy on devset	119h	18.7 %
Scribosermo	above model	314h	11.7 %
IMS-Speech [1]	mixed dataset	806h	12.0 %

that training results can be improved by a larger margin if the training is run again with the same parameters, but using the current model checkpoint as initialization model. This follows the ideas of *Stochastic Gradient Descend with Restart* [19], but uses the already implemented early-stopping with learning rate reductions on plateaus approach instead of a cosine annealing learning rate.

Table 3: *Testing different training setups.*

Notes	Duration	WER
full CV trainset	720h	7.5 %
Iteration 1	314h	8.3 %
Iteration 2	314h	7.8 %
Iteration 3	314h	7.7 %

4.3. Training in other languages

Scribosermo's approach is competitive in other languages like Spanish and French as well, which is shown in Table 4. To simplify the transfer-learning process the usual two-step frozen and unfrozen training with a reinitialized last layer was replaced with a simpler approach that does not require freezing of parts of the network. First, the alphabet size of the two languages was reduced, using the rules for cross-word puzzles, which replace letters that contain diacritics and ligatures with their basic form. Using the cross-word puzzles rules has the advantage that they are commonly known and therefore should not pose a problem for humans reading the predicted transcriptions. Following this approach, the French alphabet now has the same letters as the English, only the Spanish has an extra letter (ñ). Thus, the size of the last layer for Spanish still has to be changed, but instead of completely reinitializing the new layer, it is only extended with new weights for the extra letter. Thereby the pretrained English weights for the other letters can be kept, which greatly improves the results, similar to only training over the pretrained weights in German. A future optimization step could include replacing phonetically similar but otherwise different characters in the base alphabet with ones from the target alphabet, which was explored in more depth by [20] for training-free language adoption.

To compare the influence of the alphabet extension, a separate experiment was run following the usual training approach of reinitializing the complete last layer for the larger Spanish alphabet. The first part of Table 5 shows that the single-step approach with alphabet extension performs better than a simple single-step training with reinitialization of the last layer, as the deeper layers' weights are not so much influenced by backpropagation of prediction errors coming from the random weights

Table 4: *Spanish and French training results on CommonVoice testset. Above is Spanish, below is French.*

	Notes	Duration	WER
Scribosermo	using CV v6	203h	10.9 %
VoxPopuli [3]	using CV v5	203h	9.6 %
Luo et al. [5]	greedy on devset	96h	15.0 %
Scribosermo	using CV v6	364h	12.5 %
VoxPopuli [3]	using CV v5	364h	10.0 %

of the last layer at the beginning of the training. Compared to the more usual two-step training which solves this problem it is much faster (trainings were executed on 2× Nvidia-V100).

Similar to extending the last layer of the network for new alphabet letters, it is also possible to drop characters to reduce the alphabet size. Following the cross-word puzzle approach, the converted Italian alphabet is the same as the English one. But as Italian sounds more similar to Spanish than to English, it is beneficial to use a Spanish network to train upon, after dropping the extra Spanish letter (Table 5, second part).

Table 5: *Influence of finetuning with alphabet extension on Spanish (above) and alphabet shrinking for Italian (below).*

Notes	WER	Traintime
single-step reinitialization	11.66 %	18h
two-step training	11.11 %	13+18h
alphabet extension	11.05 %	19h
Notes	**Duration**	**WER**
English → Italian	111h	13.8 %
Spanish → Italian	111h	12.2 %

4.4. Inference speed

The main benefit of a small network is fast inference speed on low powered devices. This usually comes with a trade-off of a loss in recognition accuracy as larger models can store more information in their weights, but a fast model that can run even on devices with low computation capabilities is a key feature of this work. The full model itself has a size of about 75MB, the quantized model about 20MB.

The transcription speed is evaluated in Table 6, which shows that the presented models are much faster than the model of *IMS-Speech*, and that they can run faster than real-time on a RaspberryPi. To reduce the memory requirements for very long inputs, they can also be transcribed chunk by chunk in a streaming manner. Here the full CTC-labels were calculated and afterwards given as input to the decoder, similar as in *IMS-Speech*. The authors of *VoxPopuli* did not publish the inference speed of their network. A comparison of the network parameter count between their *wav2vec-base* net, which has about 95M params, and *QuartzNet15x5* which only has 19M, allows an estimation that it might run about 5x slower.

Table 6: *Inference Speed, measured as Real Time Factor*

Device	Model	RTF
PC - 1 core AMD3700X		0.24
PC - 1 core (unknown)	net of *IMS-Speech* [1]	14.2
RaspberryPi-4 - 4gb	tflite full	1.3
RaspberryPi-4 - 4gb	tflite optimized (TLO)	0.7
RaspberryPi-4 - 4gb	*DeepSpeech* TLO	0.7

4.5. Training with all datasets

After demonstrating the performance of the presented approach with relatively small datasets, an experiment was run to measure the influence of larger datasets on the transcription performance. In total 37 datasets for German [2, 4, 21–55], 8 datasets for Spanish [4, 21, 24, 27, 29, 32, 55, 56], 7 datasets for French [4,21,24,27,29,32,55] and 5 datasets for Italian [4,21,27,29,55] were collected. The trainings were continued with the models of the trainings with CommonVoice only, and afterwards the models were finetuned on this dataset again. The results can be found in Table 7 and show that the advantage of using more data is relatively small. Possible explanations might be that the quality of the mixed datasets is not very good or differs too much from the test recordings, or that the small network is reaching its maximum information capacity.

Table 7: *Training with all accessible datasets in German (DE), Spanish (ES), French (FR) and Italian (IT). Datasets for testing are either CommonVoice (CV) or Tuda (TD).*

Language	#Datasets	Duration	WER
DE-CV	37	2370 h	6.6 %
DE-TD			10.2 %
ES-CV	8	817 h	10.0 %
FR-CV	7	1028 h	11.0 %
IT-CV	5	360 h	11.5 %

5. Corcua

In this chapter the library which was built to handle the above datasets is presented. Often speech-to-text frameworks like Mozilla's *DeepSpeech* or Nvidia's *NeMo* have customized scripts for downloading and converting a range of supported datasets into their custom dataset format. But many datasets are used in multiple frameworks, so large parts of the scripts have overlapping tasks. The *audiomate* [57] library was built to ease the use of different audio datasets for machine learning tasks and is able to load 18 different datasets and export them into 4 different speech-to-text frameworks. But extending it with new datasets is quite complicated and requires a deeper understanding of the architecture. The goal in creating *corcua* was not only to build a library to load different audio datasets and export them to different framework formats, but also to make adding new datasets as easy as possible.

Corcua's architecture is split into three different parts, *downloading*, *reading* and *writing*. The *downloader's* task is to download and extract a dataset to a local directory. Helper functions for common formats like *zip* and *tar.gz* or downloading from a server directory are already pre-implemented. A *reader* loads the audio files, transcriptions and optionally other information included in the dataset into an easy to handle dictionary format and returns a list of items like this:

```
item = {
  "filepath": "path/to/audiofile",
  "speaker": "Newton",
  "text": "That damn apple!"
}
```

A *writer* takes a list of dictionaries and saves them into the requested framework's dataset format, like *csv* or *json*. It also converts the audio files from different codecs to the commonly used *wav* encoding.

Besides mere dataset processing, there are also tools to print some statistics about the dataset, like total duration or the most recorded speakers. *Corcua* also supports splitting datasets into different partitions, like train and test, either randomly or by key separated classes, for example that all utterances of one speaker are in the same partition.

Compared to *audiomate*, conversions of some datasets are much faster. Converting the German CommonVoice-v5 dataset (701 h) with *audiomate* took about 12 h on a modern CPU, while converting the slightly larger CommonVoice-v6 dataset (777 h) with *corcua* takes less than 3 h.

Currently *corcua* can load 34 different datasets (18 of them are German only, 13 are available in more than three languages), and is able to write them into 3 framework formats. Some of the multilingual datasets have been extracted from computer games like *Skyrim* or *The Witcher*, which often provide high quality dialogs. With the included support of extracting labels from manually transcribed YouTube videos, it is possible to create audio datasets for almost any language. *Corcua* has been released as open source project and can be accessed under: *https://gitlab.com/Jaco-Assistant/corcua*

6. Conclusion

In this paper small Speech-to-Text models for German, as well as for Spanish, French and Italian, were presented. The models combine the advantages of other approaches. They are competitive with the best models to date on the CommonVoice dataset in German, Spanish and French, as well as with the best one on the German Tuda dataset. At the same time they can run in real-time on single-board computers like a RaspberryPi and can be trained on consumer-grade hardware with a comparatively small dataset. These models are especially interesting for embedded or offline speech applications, for example in smart home systems running on edge-devices with low power consumption, or on smartphones in environments where no stable internet connection is available. Running offline on standard hardware also has advantages if users do not want and companies are not allowed to use cloud providers for privacy reasons.

7. References

[1] P. Denisov and N. T. Vu, "Ims-speech: A speech to text tool," *arXiv preprint arXiv:1908.04743*, 2019.

[2] S. Radeck-Arneth, B. Milde, A. Lange, E. Gouvêa, S. Radomski, M. Mühlhäuser, and C. Biemann, "Open source german distant speech recognition: Corpus and acoustic model," in *International Conference on Text, Speech, and Dialogue*. Springer, 2015, pp. 480–488.

[3] C. Wang, M. Rivière, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation," *arXiv preprint arXiv:2101.00390*, 2021.

[4] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[5] J. Luo, J. Wang, N. Cheng, E. Xiao, J. Xiao, G. Kucsko, P. O'Neill, J. Balam, S. Deng, A. Flores *et al.*, "Cross-Language Transfer Learning and Domain Adaptation for End-to-End Automatic Speech Recognition," in *2021 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2021, pp. 1–6.

[6] Mozilla, "Project DeepSpeech," 2021, [accessed 26-February-2021]. [Online]. Available: https://github.com/mozilla/DeepSpeech

[7] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[8] J. Park, Y. Boo, I. Choi, S. Shin, and W. Sung, "Fully neural network based speech recognition on mobile and embedded devices," in *Advances in neural information processing systems*, 2018, pp. 10 620–10 630.

[9] Y. Zhang, S. Sun, and L. Ma, "Tiny transducer: A highly-efficient speech recognition model on edge devices," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6024–6028.

[10] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6381–6385.

[11] A. Ghoshal, P. Swietojanski, and S. Renals, "Multilingual training of deep neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 7319–7323.

[12] S. Thomas, S. Ganapathy, and H. Hermansky, "Multilingual MLP features for low-resource LVCSR systems," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 4269–4272.

[13] S. Radeck-Arneth, B. Milde, A. Lange, E. Gouvêa, S. Radomski, M. Mühlhäuser, and C. Biemann, "Open source german distant speech recognition: Corpus and acoustic model," in *International Conference on Text, Speech, and Dialogue*. Springer, 2015, pp. 480–488.

[14] ACL, "ACL 2013 EIGHTH WORKSHOP ON STATISTICAL MACHINE TRANSLATION," 2013, [accessed 23-March-2021]. [Online]. Available: https://www.statmt.org/wmt13/translation-task.html

[15] MozillaItalia, "DeepSpeech-Italian-Model: Mitads," 2021, [accessed 28-June-2021]. [Online]. Available: https://github.com/MozillaItalia/DeepSpeech-Italian-Model/releases/tag/Mitads-1.0.0-alpha2

[16] D. Povey, "Poco LM," 2021, [accessed 29-June-2021]. [Online]. Available: https://github.com/danpovey/pocolm

[17] S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang, "Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6124–6128.

[18] Nvidia, "NeMo," 2021, [accessed 26-February-2021]. [Online]. Available: https://github.com/NVIDIA/NeMo

[19] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.

[20] M. Prasad, D. van Esch, S. Ritchie, and J. F. Mortensen, "Building Large-Vocabulary ASR Systems for Languages Without Any Audio Training Data." in *INTERSPEECH*, 2019, pp. 271–275.

[21] VoxForge, "VoxForge," 2021, [accessed 23-March-2021]. [Online]. Available: http://www.voxforge.org/

[22] B. A. for Speech Signals, "FORMTASK," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0005-8535-9

[23] ——, "SprecherInnen," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0003-FF39-F

[24] K. Park and T. Mulc, "CSS10: A Collection of Single Speaker Speech Datasets for 10 Languages," *Interspeech*, 2019.

[25] s. Piranha Bytes, THQ Nordic GmbH, "Gothic 1-3," 2001, [accessed 23-March-2021]. [Online]. Available: https://www.gog.com/game/gothic

[26] Kurzgesagt, "Kurzgesagt," 2021, [accessed 23-March-2021]. [Online]. Available: https://www.youtube.com/c/KurzgesagtDE/videos

[27] LinguaLibre, "LinguaLibre," 2021. [Online]. Available: https://lingualibre.org/wiki/LinguaLibre:Main_Page

[28] Musstewissen, "Musstewissen Deutsch, Mathe, Physik, Chemie," 2021, [accessed 23-March-2021]. [Online]. Available: https://www.youtube.com/c/musstewissenDeutsch/videos,https://www.youtube.com/c/musstewissenMathe/videos,https://www.youtube.com/c/musstewissenPhysik/videos,https://www.youtube.com/c/musstewissenChemie/videos

[29] M-AILABS, "The M-AILABS Speech Dataset," 2019, [accessed 23-March-2021]. [Online]. Available: https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/

[30] PULS, "PULS Reportage," 2021, [accessed 23-March-2021]. [Online]. Available: https://www.youtube.com/puls/videos

[31] A. Köhn, F. Stegen, and T. Baumann, "Mining the Spoken Wikipedia for Speech Data and Beyond," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, N. C. C. Chair), K. Choukri, T. Declerck, M. Grobelnik, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, Eds. Paris, France: European Language Resources Association (ELRA), may 2016.

[32] Tatoeba, "Tatoeba," 2021, [accessed 23-March-2021]. [Online]. Available: https://tatoeba.org/eng/

[33] ZDF, "TerraX," 2021, [accessed 23-March-2021]. [Online]. Available: https://www.youtube.com/c/terra-x/videos

[34] Y-Kollektiv, "Y-Kollektiv," 2021, [accessed 23-March-2021]. [Online]. Available: https://www.youtube.com/c/ykollektiv/videos

[35] Goofy, "Zamia-Speech," 2021, [accessed 23-March-2021]. [Online]. Available: https://goofy.zamia.org/zamia-speech/corpora/zamia_de/

[36] B. A. for Speech Signals, "Alcohol Language Corpus," 2016, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0001-88E5-3

[37] ——, "BROTHERS," 2016, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0001-55C3-3

[38] ——, "HEMPEL," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0002-F80E-8

[39] ——, "PHATTSESSIONZ," 2016, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0000-CC6A-4

[40] ——, "PhoneDat1," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0001-D20B-6

[41] ——, "RVG1," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0004-3FF4-3

[42] ——, "RVG-J," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0004-AE1D-9

[43] ——, "SC10," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0002-1129-D

[44] ——, "SHC," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0007-0700-1

[45] ——, "SI100," 2020, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0007-E9CF-A

[46] ——, "SMC," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0005-C50F-D

[47] ——, "VM1," 2016, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0000-EB31-0

[48] ——, "VM2," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0000-FC55-5

[49] ——, "WaSeP," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0007-3D30-F

[50] ——, "ZIPTEL," 2017, [accessed 23-March-2021]. [Online]. Available: http://hdl.handle.net/11022/1009-0000-0003-1E02-A

[51] T. Müller, "Thorsten Müller (TTS)," 2021, [accessed 23-March-2021]. [Online]. Available: http://www.openslr.org/95/

[52] R. . T. N. GmbH, "Guild2-Renaissance," 2010, [accessed 23-March-2021]. [Online]. Available: https://www.gog.com/game/the_guild_2_renaissance

[53] Bethesda, "Skyrim-Legacy+DLCs," 2011, [accessed 23-March-2021]. [Online]. Available: https://store.steampowered.com/app/72850/The_Elder_Scrolls_V_Skyrim/

[54] C. P. RED, "Witcher3-GOTY," 2016, [accessed 23-March-2021]. [Online]. Available: https://www.gog.com/game/the_witcher_3_wild_hunt_game_of_the_year_edition

[55] E. Salesky, M. Wiesner, J. Bremerman, R. Cattoni, M. Negri, M. Turchi, D. W. Oard, and M. Post, "Multilingual TEDx Corpus for Speech Recognition and Translation," 2021.

[56] carlfm01, "Librivox-Spanish," 2019, [accessed 23-March-2021]. [Online]. Available: https://www.kaggle.com/carlfm01/120h-spanish-speech

[57] M. Büchi and A. Ahlenstorf, "audiomate: A Python package for working with audio datasets," *Journal of Open Source Software*, vol. 5, no. 52, p. 2135, 2020.