Swiss German Speech to Standard German Text

SwissText.org Shared Task 3

Alex Wolf
University of Zurich
alex.wolf@uzh.ch

Deborah Noemie Jakobi
University of Zurich
deborahnoemie.jakobi@uzh.ch

Abstract

This paper presents a solution to the SwissText Conference (STC) shared task 3, 2021. The shared task challenges participants to implement models for translating Swiss German speech to standard German text. The authors implemented multiple *DeepSpeech* models using the data provided by SwissText.org as well as additional data which lead to improvement of the model. An additional experiment with a sequence to sequence translation model was trained in order to improve our score. We achieved a BLEU score of up to 0.17 on the test set of SwissText. Although the performance of the final model did not score well in comparison with other submissions, there is plenty of room for possible improvements.

1 Introduction

Automatic speech recognition (ASR) means translating a spoken utterance into written text. It can, for example, be used for voice assistants or automatic transcription of audio or video files. While pre-trained Speech-to-Text (STT) models for English are available, German models or even Swiss German models are rare or not existing (Agarwal and Zesch, 2019). Swiss German has a wide variety of different Swiss German dialects, with a huge difference in words, pronunciation, even to the point of sounding like a different language. Swiss German has relatively few speakers (around 5 million) and there is hardly any standardized spelling. This leads to Standard German being one of the official writing languages in Switzerland. As there is no official Swiss German spelling, most speakers using written Swiss German just use their own spelling, which resembles mostly a phonetic translation. This leads to huge variance within Swiss German writing (Pluss et al., 2020). Therefore, it makes sense to translate spoken Swiss German to Standard German in order to correspond to the official language situation. Tackling a standardized translation of different spoken Swiss German dialects into standardized German text requires a vast amount of data and fine-tuning. The STC 2021 proposed a shared task to tackle this problem and provided a dataset (SwissText Conference dataset (STCD) to train and fine-tune on. This paper shows what kind of experiments, data, and approaches the authors used to tackle this problem. The proposed task is very complex as it includes not only a STT conversion but also translation from Swiss German to Standard German which can be referred to as speech translation (Pluss et al., 2020). Additionally, it possibly includes domain shift, as the training data stems only from Swiss parliament speeches while the domain of the test set is unclear. This paper presents an overview of the previous research done within this area, an introduction to the DeepSpeech model used for tackling the task, and our experiments and results.

2 Literature review

The shared task of translating spoken Swiss German into standard written German was already presented by the STC in 2020. The only difference between the tasks is that this year's task provides more data, while last year's task was specifically about low-resource languages. While the current 2021 task is about reaching the highest BLEU score, last year's submissions were ranked based on the least Word Error Rate (WER). Buchi et al. (2020) achieved the best WER of 40.29%. The authors used a CNN acoustic model named Jasper. They used additional Standard German data and fine-tuned on the official data set. They used different augmentation techniques and a language model. (Kew et al., 2020) achieved the second-best WER of 45.45%. They used a DNN-HMM time-delay neural model including a specifically created pro-

nunciation lexicon.
Agarwal and Zesch (2020) used an end-to-end model called DeepSpeech and achieved an WER of 58.93%. This is the model we decided to use as well. It is publicly available on GitHub and can easily be fine-tuned or used for transfer learning Pluss et al. (2020).

3 DeepSpeech

Mozilla DeepSpeech is an end-to-end STT model using tensorflow. It was first developed for translating English speech to English text (Hannun et al., 2014). It is implemented using machine translation techniques. Agarwal and Zesch (2019) have implemented a German STT model using DeepSpeech. They provide all of their code and pre-trained models including ready-made scripts for transfer-learning and fine-tuning (Agarwal and Zesch, 2019). This avoids privacy issues of common web-services that require uploading potentially private data. Additionally, researchers are free to adjust and extend the model according to their requirements. DeepSpeech is a deep recurrent neural network (RNN) on character level. It can be trained using supervised learning.

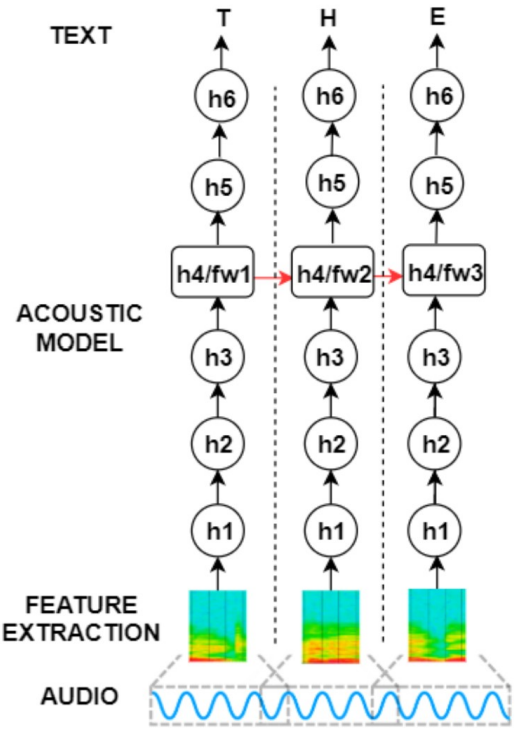

Figure 1: DeepSpeech architecture according to Agarwal and Zesch (2019)

The DeepSpeech architecture, as shown in figure 1, consists of 6 layers. The features extracted from the audio files are Frequency Cepstral Coefficients. Those essentially capture the frequency spectrum of an audio file in a condensed format. Those are fed to the first layer which is a fully connected, i.e. dense, layer like the next two. The fourth layer consists of unidirectional feed forward layer. The fifth layer is another dense layer which is followed by the output layer. A more in depth description can be found in Agarwal and Zesch (2019). As becomes clear, the model has no additional phoneme-to-grapheme model but directly outputs the transcribed characters, respectively their probabilities. The Connectionist Temporal Classification (CTC) loss function is used to maximize the probability of the output characters. This loss is specifically designed for tasks where the prediction categories have unclear boundaries. In this case characters which can refer to phonemes that span over times frames of various length.
The German DeepSpeech model integrates a language model that has to be built first. We trained it on two publicly available German datasets: One released by the University of Hamburg containing 8 million sentences [1] and the other the Europarl corpus containing another 2 million sentences [2].

4 Experiments

4.1 Training Data

To train our model we use a subset of the dataset provided by the STC as well as the ArchiMob corpus. The STC provides two different datasets. An unlabelled dataset of about 300 hours spoken Swiss German. The labeled dataset contains mostly the Bernese dialect. An additional 1200 hours of unlabeled spoken Swiss German audio data. The unlabelled dataset includes mostly the Zurich dialect. Due to time limitations, we did not use the second provided dataset. We remove all audio files from the STCD where the quality of the translation is rated lower than 0.7 (rating provided by STC) which are still around 130k files. Additionally, we also need to remove files larger than 1 MB due to memory limitations (ca. 7k files). All files were converted from FLAC to Wav files and the sample rate was reduced from 48'000 Hz to 16'000 Hz to match the DeepSpeech input requirements. The

[1] UHH Open source acoustic models - https://www.inf.uni-hamburg.de/en/inst/ab/lt/resources/data/acoustic-models.html
[2] Europarl - https://www.statmt.org/europarl/

Model#	Data	Train BLEU	Test BLEU	WER
1	SwissText	0.23	0.0004	-
2	ArchiMob	0.27	**0.17**	0.52
3	ArchiMob + SwissText	0.28	0.1	0.54
4	ArchiMob + SwissText + text-to-text	0.16	0.07	-

Table 1: Results

official test set contains 13 hours of data (Ando and Zhang, 2005) and has a dialect distribution conforming to the actual Swiss German dialect distribution. In addition to the official datasets, we used the ArchiMob corpus (Release 2) which contains 57 hours of spoken Swiss German data (Samardžić et al., 2016). They provide Swiss German and German transcriptions and it consists of various different dialects. The German DeepSpeech comes with a script to pre-process the transcriptions. The pre-processing includes:

- removing of all unallowed characters (allowed are a-zA-Zäüö)
- convert special characters like $ to their written form (i.e. dollar)
- convert numbers to their written form
- lowercase
- map all character with diacritics to one of the allowed characters

This fits the requirements given by the STC.

4.2 Evaluation metrics

We evaluate our model through two types of metrics. The BLEU score (Papineni et al., 2002), which is required by the STC to compare the results. The STC specifically requires a corpus-based BLEU score, which aims at measuring the distance/similarity of the generated text and the provided ground truth. As the German DeepSpeech test script does not include the BLEU score we added it manually.

Additionally, we keep track of the WER for our models, as this is a common metric for speech recognition systems (Park et al., 2008) which is also included in the German DeepSpeech.

4.3 Environment

We train the model on a single NVIDIA GeForce RTX 2070 Laptop GPU (8 GB memory). We perform minimal tuning of our model's hyperparameters following the work of Agarwal and Zesch (2020). .

4.4 Models

Following Agarwal and Zesch (2020) we use a DeepSpeech architecture (Hannun et al., 2014) as our main model for speech-to-text translation. In order to get better results we use a pre-trained DeepSpeech model (Ohme, 2020) as the base model for most of our experiments. The first model we trained served as our private baseline.

Baseline model We trained a bare DeepSpeech model with the default DeepSpeech hyperparameters on the labeled STCD which achieved a BLEU score of 0.13 on our internal test set. The baseline model will be referred to as the baseline model or model #0 henceforth.

Pre-trained model #1 In order to improve our baseline model we use a pre-trained German DeepSpeech model (Ohme, 2020), which is fine-tuned on the STCD dataset. We used Agarwal and Zesch (2020) best-reported hyperparameters which are learning rate of 0.0001, dropout of 0.25. The alpha and beta values for the language model are 0.40 and 1.10 respectively. We continue to use these hyperparameters as those provided better results. The model achieved a BLEU score of 0.23 on the validation set and 0.0004 on the official test set.

ArchiMob data model #2 As the 1st model did not achieve the expected results we decided to fine-tune our model on the ArchiMob data, as we expected the data distribution to be closer to the actual test set. We trained for another 30 epochs with the same hyperparameters as in model #1. We achieved the following BLEU scores for the validation and test set, 0.27, 0.17, respectively.

Augmented data model #3 In order to produce a more robust model, that can handle different pronunciations and speeds, we added data augmentation to our 2nd model. We fine-tuned our model

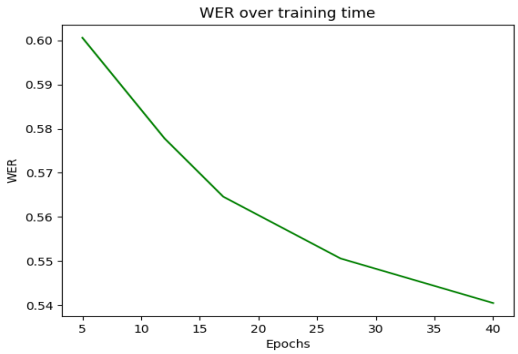

Figure 2: WER per epoch

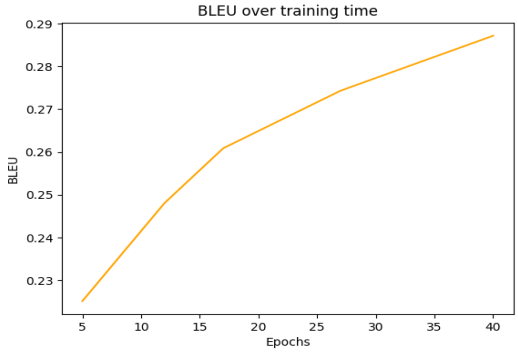

Figure 3: BLEU per epoch

with the following data augmentations:

- warp: "Applies a non-linear image warp to the spectrogram. This is achieved by randomly shifting a grid of equally distributed warp points along the time and frequency axis" (DeepSpeech, 2020)
- tempo: "Scales spectrogram on the time axis and thus changes playback tempo." (DeepSpeech, 2020)

The STCD dataset is used for the augmentation fine-tuning on the 3rd model. This approach achieved a BLEU score of 0.28 on the validation set and 0.1 on the test set. The model was trained for another 20 epochs with a warp probability of 0.1 and a tempo probability of 0.1. One epoch on this model took around 4h to complete. Due to time limitations we were not able to continue training the model any further, but the model was still improving and we expect that this approach could produce even better results. The WER and BLEU progression are shown in figures 2 and 3.

Text-to-Text model #4 Our 4th approach tries to improve the output by adding a text-to-text translation model after the speech-to-text model. We fine-tuned a pre-trained German to English model following Tiedemann and Thottingal (2020) using the "huggingface" framework (Wolf et al., 2020) on the STCD dataset. With this approach we achieved a BLEU score of 0.13 on the validation dataset and 0.07 on the test dataset.

5 Results & Discussion

Interestingly, we noticed that our internal deviations in the BLEU score are not as great as on the official test set. The BLEU score, as well as the WER values, are shown for each model in table 1.

After multiple improvements, our best results were achieved by our 2nd model. We assume that this is due to the dialect distribution on the actual test set, which is closer to the ArchiMob dataset than the STCD dataset. We expect that further improvements can be achieved by the data augmentation approach, which had to be stopped due to time limitations. As previously explained in the model description the 4th model was still improving and should be trained more extensively.

6 Conclusion

Although the other submissions to the task provided better results our model showed a constant increase in performance over the training time. Due to time limitations, we had to stop training before our model has reached a point where it stopped improving. Overall, the DeepSpeech model provides a neat end-to-end architecture that provides a solid foundation for fine-tuning and adaptation to various languages. Further improvements could be achieved by self-training on the unlabelled data, which would allow us to incorporate more data into our training process. This approach would increase the amount of available data. While most likely providing better results and being less expensive than manually labeling the data. Another approach would be to add a better language model to rate the sentences provided by the beam search outputs. In addition to the two approaches, we expect that more exhaustive training with data augmentation could greatly improve the model. Apart from longer training time, a hyperparameter tuning approach with more augmentation methods could provide more insight into the topic.

References

Aashish Agarwal and Torsten Zesch. 2019. German end-to-end speech recognition based on deepspeech. In *KONVENS*.

Aashish Agarwal and Torsten Zesch. 2020. Ltl-ude at low-resource speech-to-text shared task: Investigating mozilla deepspeech in a low-resource setting. In *SwissText/KONVENS*.

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Matthias Buchi, Malgorzata Anna Ulasik, Manuela Hürlimann, Fernando Benites, Pius von Daniken, and Mark Cieliebak. 2020. Zhaw-init at germeval 2020 task 4: Low-resource speech-to-text.

DeepSpeech. 2020. Training your own model — deepspeech 0.9.3 documentation.

Awni Y. Hannun, Carl Case, J. Casper, Bryan Catanzaro, G. Diamos, Erich Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Ng. 2014. Deep speech: Scaling up end-to-end speech recognition. *ArXiv*, abs/1412.5567.

Tannon Kew, Iuliia Nigmatulina, Lorenz Nagele, and Tanja Samardžić. 2020. Uzh tilt: A kaldi recipe for swiss german speech to standard german text.

Karsten Ohme. 2020. Pre-trained deepspeech model v0.9.0.

Kishore Papineni, S. Roukos, T. Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Youngja Park, Siddharth Patwardhan, K. Visweswariah, and S. C. Gates. 2008. An empirical analysis of word error rate and keyword error rate. In *INTERSPEECH*.

Michel Pluss, Lukas Neukom, and Manfred Vogel. 2020. Germeval 2020 task 4: Low-resource speech-to-text.

Tanja Samardžić, Yves Scherrer, and Elvira Glaser. 2016. ArchiMob - a corpus of spoken Swiss German. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4061–4066, Portorož, Slovenia. European Language Resources Association (ELRA).

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT — Building open translation services for the World. In *Proceedings of the 22nd Annual Conferenec of the European Association for Machine Translation (EAMT)*, Lisbon, Portugal.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.