UZH_TILT: A Kaldi recipe for Swiss German Speech to Standard German Text

Tannon Kew
Iuliia Nigmatulina
Lorenz Nagele
Tanja Samardžić
University of Zurich
iuliia.nigmatulina, tannon.kew, lorenz.nagele, tanja.samardzic@uzh.ch

Abstract

Swiss German Speech-to-Text (STT) is a challenging task due to the fact that no single-dominant pronunciation or standardised orthography exists. This is compounded by a severe lack of appropriate training data. One potential avenue, and that which is investigated as part of the GermEval 2020 Task 4 on Low-Resource Speech-to-Text, is to translate spoken Swiss German into standard German text implicitly through STT. In this paper, we describe our proposed system that makes use of the Kaldi Speech Recognition Toolkit to implement a time delay neural network (TDNN) Acoustic Model (AM) with an extended pronunciation lexicon and language model. Using this approach, we achieve a word error rate of 45.45% on the held-out test set.

1 Introduction

In this paper, we describe our approach for the GermEval 2020 Task 4 on Low-Resource Speech-to-Text (Plüss et al., 2020) held as part of the 5th SwissText and the 16th KONVENS Joint Conference 2020. The goal of this shared task is to develop a STT system capable of converting Swiss German speech utterances into standard German text.

Our system makes use of the Kaldi Speech Recognition Toolkit (Povey et al., 2011). Specifically, we adapt the WSJ *chain* recipe to integrate a time delay neural network (TDNN) component in the process of training the acoustic model (AM) with iVectors (Peddinti et al., 2015). The TDNN architecture allows for better learning of long term temporal dependencies between phonemes in a sequence. Using iVectors potentially contributes to better generalisation to unseen data, and to DNN adaptation with the additional feature normalisation (Saon et al., 2013; Miao et al., 2015). In addition to the data set provided by the organisers, we use an external pronunciation lexicon (Schmidt et al., 2020) and the German section of the Sparcling corpus[1] (Graën et al., 2019) to build a robust N-gram language model, suitable for the target domain.

The layout of this report is as follows: Section 2 describes the aim of the shared task and the data provided. In Section 3, we describe our approach and the individual components used in our system. We report the overall performance of our system based on a held-out development set and the task test set in Section 4. Finally, in Section 5, we conclude with a discussion on some of the advantages and limitations of our approach.

2 Data

The dataset for this shared task was provided by the organisers and comprises a training set of approximately 70 hours of annotated speech data from Swiss parliamentary discussions plus an additional 4 hours of audio recordings for system evaluation. This test set only contains recordings of speakers that are not present in the training data.

The training data includes a total of 36,572 utterances spoken by 191 different speakers. Each utterance is annotated with its transcription in standard German and a unique speaker ID. According to the description of the shared task data, spoken utterances are predominantly in the Bernese dialect, with some in standard German.

We enrich the training data with two external

[1] The Sparcling corpus is described in detail as 'FEP 9' in (Graën, 2018).

Split	No. of Utterances
Train	32,916
Dev	3,656
Test	2,014

Table 1: Distribution of the datasets.

sources. First, we derive a high-coverage pronunciation lexicon containing more than 38,000 standard German words with an approximate Swiss German pronunciation to facilitate AM training. Second, we add to the N-gram language model (LM) trained on the shared task data an additional 4-gram LM trained on the German section of the Sparcling corpus (Graën et al., 2019). These steps are described in more detail in Sections 3.2 and 3.3, respectively.

2.1 Preprocessing

Utterance transcriptions are already partially preprocessed with character mapping to a defined set of allowable characters and lowercasing applied[2]. Therefore, we only apply one further step for text preprocessing, namely tokenisation. We use a simple, general-purpose tokeniser trained on German from the Python NLTK module[3].

Once tokenised, we set aside 10% of the training data as a development set for the purpose of fine-tuning model parameters. Table 1 gives an overview of the dataset splits used for this task.

3 Methods

In this section, we present the main components of our STT system, namely, the acoustic model, pronunciation lexicon and language model.

3.1 Acoustic Model

We base our STT system for Swiss German on the the WSJ *chain* recipe with the time delay neural network (TDNN) architecture provided in the Kaldi toolkit. The alignment between acoustic signal segments and transcriptions is attained with the GMM-HMM discriminative model trained with a Maximum Mutual Information criterion (MMI) with 4,000 senones and 40,000 Gaussians.

We use 13-dimensional Mel-Frequency Cepstral Coefficients (MFCC) features with cepstral mean-variance normalisation (CMVN), the first and second derivatives, and Linear Discriminative Analysis (LDA) and Maximum Likelihood Linear Transform (MLLT) transformations. In addition, we include 100-dimensional iVectors extracted from each speech frame in order to normalise the variation between speakers and dialectal varieties.

To increase the amount of training data and improve robustness of the AM, we perform popular data augmentation techniques, such as audio speed perturbation with speed factors of 0.9, 1.0, 1.1, followed by volume perturbation with volume factors sampled from the interval $[0.125, 2.0]$ (Ko et al., 2015).

The AM was trained with NVIDIA Tesla K80 GPUs and took around 14 hours.

3.2 Pronunciation Lexicon

For the pronunciation lexicon, we make use of an 11,000 word dictionary mapping standard German words to their Swiss German pronunciations (Schmidt et al., 2020). This dictionary contains manually annotated pronunciation strings (in the SAMPA alphabet (Wells et al., 1997)) for six major regional varieties, namely Zurich, St. Gallen, Bern, Basel, Valais and Nidwalden. Since the task data predominantly consists of Bernese dialect, we use the pronunciations strings for this regional variety only. Furthermore, we normalise the standard German words using the same text preprocessing steps as provided in the shared task description (i.e. character mapping and converting to lowercase).

Initially, the SAMPA dictionary provides only 15% lexical coverage of the shared task dataset. In order to increase this, we train a transformer-based grapheme-to-phoneme (g2p) model[4] on the available pairs (standard German, Swiss SAMPA) and apply it on the words from the dataset for which manual Swiss SAMPA annotation is missing. We train the g2p model with the default settings.[5]

As a result of this process, we attain a lexicon that provides 97.5% coverage of the shared task dataset. The remaining 2.5% of items not covered in the extended lexicon include tokens consisting of digits (e.g. numbers, dates, etc.) and punctua-

[2] https://www.cs.technik.fhnw.ch/speech-to-text-labeling-tool/swisstext-2020/competition/1

[3] https://www.nltk.org/api/nltk.tokenize.html#module-nltk.tokenize.toktok

[4] https://github.com/cmusphinx/g2p-seq2seq

[5] Default settings for g2p-seq2seq are as follows: size of each hidden layer = 256, number of layers = 3, size of the filter layer in a convolutional layer = 512, number of heads in multi-attention mechanism = 4.

AM	LM	Dev	Test
TDNN-iVector	STLM+SparcLM	43.69	45.45

Table 2: WER results attained on a held-out development set and 50% of the test set.

tion (e.g. web addresses) since the original SAMPA dictionary does not contain such characters. While the overall word-level accuracy of the g2p model, estimated on a held out test set, is only 39%, the output of the model is still useful for the STT system since it provides good coverage with plausible g2p mappings confirmed by manual inspection of the output.

3.3 Language model

The language modeling component used in our system is a statistical N-gram backoff LM. We train two 4-gram LMs with interpolated modified Kneser-Ney smoothing (Chen and Goodman, 1999) using the MITLM toolkit[6] (Hsu and Glass, 2008) and combine them using linear interpolation. The first LM is estimated on the basis of our training data split (see Table 1). For simplicity, we refer to this model as the shared task LM (STLM). While this LM ensures that we capture the domain of the shared task data well, it is limited in terms of size and vocabulary. In order to improve the robustness of our system, we incorporate additional language data by estimating a second 4-gram LM on the German section of the Sparcling corpus (Graën et al., 2019).

The Sparcling corpus is a cleaned and normalised version of the Europarl corpus (Koehn, 2005), which contains a large collection of parallel texts based on debates published in the proceedings of the European Parliament. In total, the Sparcling corpus provides 1.75M German utterances which are considered to be close to the target domain. The resulting LM is too large to be used directly, so we prune it using the SRILM toolkit (Stolcke, 2002), setting a threshold of 10^{-8}. We refer to this model as the Sparcling LM (SparcLM). Once pruned, we linearly interpolate the STLM and the SparcLM with weights $\lambda = 0.7$ and $\lambda = 0.3$, respectively.

4 Results

Table 2 reports the WER results attained by our TDNN-iVector STT system (see 3.3) on the held-out development set and on the test set provided for the submission[7]: the system achieves a WER of 43.69% and 45.45%, respectively. The model was tuned with language model weights (LMWT) ranging from 7 to 17, and different word insertion penalty values (WIP) of 0.0, 0.5 and 1.0. Optimal parameters (LMWT = 9 and WIP = 0.0) were determined according to the best WER on the held-out development set and then applied in order to decode the test set for this submission.

5 Discussion

An assessment of our system output transcriptions against audio samples from the test data reveals that the results are comprehensible and depict the speech utterance well in most cases. Common errors include single missing words in the transcription, separated writing of compounds (e.g. *bildung direktor* instead of *bildungsdirektor*) and the absence of numbers. The latter can easily be explained by the fact that our lexicon does not include digits and thus needs to be further extended in order to cover such common lexical items.

We also noticed that words at the beginning and end of the audio samples are cut off in many cases, making it difficult for the system to recognise these words correctly. In addition, it is clear that speech utterances do not necessarily correspond to single sentences in many cases, but rather sentence fragments, or in some cases multiple sentences[8]. The LM, however, is trained largely on complete sentences and could thus fail to account for N-gram sequences that bridge typical sentence boundaries.

6 Conclusion

In this paper, we have described our proposed solution for the GermEval 2020 Task 4: Low-Resource Speech-to-Text challenge. We have implemented an advanced TDNN AM using popular acoustic speech data augmentation techniques available as part of the Kaldi Speech Recognition Toolkit. Our

[6] https://github.com/mitlm/mitlm

[7] This result is automatically calculated and published on the shared task's public leader board upon submission.

[8] A manual evaluation of a sample of 100 speech utterances from the test set show that 58 are not complete sentences, of which 25 also contain fragments from the preceding or following utterance.

model achieves a WER of 45.45% on the public part of the task's test set, which we believe is competitive given the amount of training data and the major challenges involved in STT for languages with a high degree of dialectal variability such as Swiss German.

Acknowledgments

We would like to thank Fransisco Campillo from Spitch AG, who helped us in setting up our initial STT system that was used as a springboard for our experiments and investigations.

References

Stanley F Chen and Joshua Goodman. 1999. An Empirical Study of Smoothing Techniques for Language Modeling. *Computer Speech & Language*, 13(4):359–394.

Johannes Graën. 2018. *Exploiting alignment in multiparallel corpora for applications in linguistics and language learning*. Ph.D. thesis, University of Zurich.

Johannes Graën, Tannon Kew, Anastassia Shaitarova, and Martin Volk. 2019. Modelling large parallel corpora: The zurich parallel corpus collection. In *Challenges in the Management of Large Corpora (CMLC-7)*. Leibniz-Institut für Deutsche Sprache.

Bo-June (Paul) Hsu and James R. Glass. 2008. Iterative language model estimation: Efficient data structure & algorithms. In *In Proceedings of the Ninth Annual Conference of the International Speech Communication Association*, pages 841–844, Brisbane, Australia.

Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. Audio augmentation for speech recognition. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

Yajie Miao, Hao Zhang, and Florian Metze. 2015. Speaker adaptive training of deep neural network acoustic models using i-vectors. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(11):1938–1949.

Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. A time delay neural network architecture for efficient modeling of long temporal contexts. In *Sixteenth Annual Conference of the International Speech Communication Association*.

Michel Plüss, Lukas Neukom, and Manfred Vogel. 2020. GermEval 2020 Task 4: Low-Resource Speech-to-Text. In preparation.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al. 2011. The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, CONF. IEEE Signal Processing Society.

George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny. 2013. Speaker adaptation of neural network acoustic models using i-vectors. In *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 55–59. IEEE.

Larissa Schmidt, Lucy Linder, Sandra Djambazovska, Alexandros Lazaridis, Tanja Samardžić, and Claudiu Musat. 2020. A swiss german dictionary: Variation in speech and writing.

Andreas Stolcke. 2002. SRILM — An Extensible Language Modeling Toolkit. In *Proceedings of the Seventh International Conference on Spoken Language Processing (ICSLP)*, Denver, USA.

John C Wells et al. 1997. Sampa computer readable phonetic alphabet. *Handbook of standards and resources for spoken language systems*, 4.