EDAudio: Easy Data Augmentation for Dialectal Audio

Lea Fischbach[1] **Akbar Karimi**[2,3] **Alfred Lameli**[1] **Lucie Flek**[2,3]

[1]Research Center Deutscher Sprachatlas, Philipps-Universität Marburg, Germany
[2]Lamarr Institute for ML and AI, Germany
[3]b-it Center, University of Bonn, Germany
lea.fischbach@uni-marburg.de

Abstract

We investigate lightweight and easily applicable data augmentation techniques for dialectal audio classification. We evaluate four main methods, namely shifting pitch, interval removal, background noise insertion and interval swap as well as several subvariants on recordings from 20 German dialects. Each main method is tested across multiple hyperparameter combinations, inlcuding augmentation length, coverage ratio and number of augmentations per original sample. Our results show that frequency-based techniques, particularly frequency masking, consistently yield performance improvements, while others such as time masking or speaker-based insertion can negatively affect the results. Our comparative analysis identifies which augmentations are most effective under realistic conditions, offering simple and efficient strategies to improve dialectal speech classification.

1 Introduction

Audio data augmentation (DA) has been widely studied, with methods typically targeting either raw waveforms (Ko et al., 2015), where speed changes are recommended for speech recognition, or spectrogram representations. Spectrogram-based approaches such as *SpecAugment* (Park et al., 2019) and *SpecMix* (Kim et al., 2021) apply image-based transformations but lack reversibility and constrain feature extraction. In contrast, raw-audio augmentation preserves full signal content and allows flexible downstream processing. Our work focuses on simple, interpretable waveform-level techniques, affecting both time and frequency dimensions.

Previous research in domains such as environmental sound classification (ESC) and automatic speech recognition (ASR) has shown varied effects of different DA methods. For instance, Salamon and Bello (2017) find pitch shifting most effective for ESC, while Fukuda et al. (2018) show speed perturbation improves ASR performance and noise addition contributes the least in both cases. However, as we demonstrate, such findings do not necessarily generalize to dialect classification, where, for example, noise addition shows stronger benefits. This highlights the need for domain-specific evaluations. We address this by adapting the four simple text-based methods from Wei and Zou (2019) to the audio domain for speech classification, evaluating their effect on dialectal audio through isolated and systematic parameter studies.

Previous research includes SpliceOut (Jain et al., 2022), which shows that removing time intervals from audio works better and is more efficient than traditional time masking. In Mixing Signals (Xu et al., 2022), the authors show that randomly mixing audio from the same class helps the model generalize better. Additionally, Braun et al. (2017) introduce Accordion Annealing, a curriculum learning method where training starts with very noisy audio (e.g., 0dB signal-to-noise ratio) and gradually includes cleaner audio, up to 50dB SNR. While some surveys, such as Ferreira-Paiva et al. (2022), cover a broad range of DA techniques, many focus on spectrograms or evaluate only combined DA setups without analyzing individual effects (Mushtaq and Su, 2020; Mushtaq et al., 2021). In contrast, our study provides a detailed, isolated evaluation of each method in the context of dialect classification.

2 Experimental Framework

This section outlines the experimental setup: used data and data preparation, classification pipeline, augmentation techniques and evaluation setup.

2.1 Data

The audio data used originates from the REDE project (Schmidt et al., 2020ff.), a large-scale project on regional linguistic variation in Germany.

Proceedings of Recent Advances in Natural Language Processing,pages 363–368
Varna, Sep 8–10, 2025
https://doi.org/10.26615/978-954-452-098-4-044

Specifically, we focus on a subset in which speakers were asked to translate 40 standard German sentences into their local dialect. To ensure strong dialectal features, only recordings from older speakers were selected. The 17.64-hour dataset consists of spoken audio from 198 speakers, covering 20 distinct German dialects classified according to Wiesinger (1983). All audio files were normalized to mono, 16-bit and 16kHz.

2.2 Classification Pipeline

First, audio files are segmented into fixed-length 10-second samples with our experimental pipeline[1]. Incomplete final segments are discarded to maintain consistency. For feature extraction, we utilize the Trillsson4 model (Shor and Venugopalan, 2022) to generate embeddings from raw audio. These serve as input to a lightweight CNN (two hidden layers with LeakyReLU + dropout, softmax output) for dialect classification.

To ensure a fair evaluation, all models are trained and validated on the same speaker-independent data split. Specifically, for each dialect, 10% of speakers are assigned to the validation set and 10% to the test set. The remaining speakers are used for training. Importantly, each speaker appears in only one of the subsets (train, validation, or test) to prevent the model from learning speaker-specific characteristics rather than dialectal features.

To account for stochastic effects, each experiment is run 50 times using fixed splits but varying initializations and data order; the median weighted F1-score over these runs is reported. Augmentation strategies were compared via a two-sided Mann-Whitney U test (Mann and Whitney, 1947) (implemented via SciPy (Virtanen et al., 2020)) on the distributions of weighted F1-scores, with Holm-Bonferroni correction (Holm, 1979) (implemented via statsmodels (Seabold and Perktold, 2010)) for multiple comparisons (α=0.05, 231 tests).

2.3 Experimental Setup

Intervals are randomly selected subparts of each 10-second segment for augmentation. To explore granularity and coverage, their size and portion are controlled by two parameters: **interval length** l_{aug} (duration of each subpart to augment) and **augmentation ratio** α (portion of the segment to modify). Given an audio segment of length l_{audio}, the number of intervals n_{aug} applied per segment is then computed as: $n_{\text{aug}} = \left\lfloor \frac{\alpha \cdot l_{\text{audio}}}{l_{\text{aug}}} \right\rfloor$.

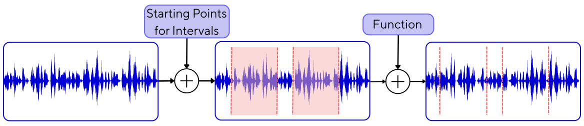


Figure 1: Example augmentation process: a 10-second segment with l_{aug}=4s and α=1.0. Two non-overlapping intervals (center) are augmented (right), leaving 2s unchanged.

A function selects n_{aug} non-overlapping intervals randomly within each segment (see Figure 1). Start positions for the intervals are sampled from a shrinking valid range to ensure disjoint placement within each 10-second segment.

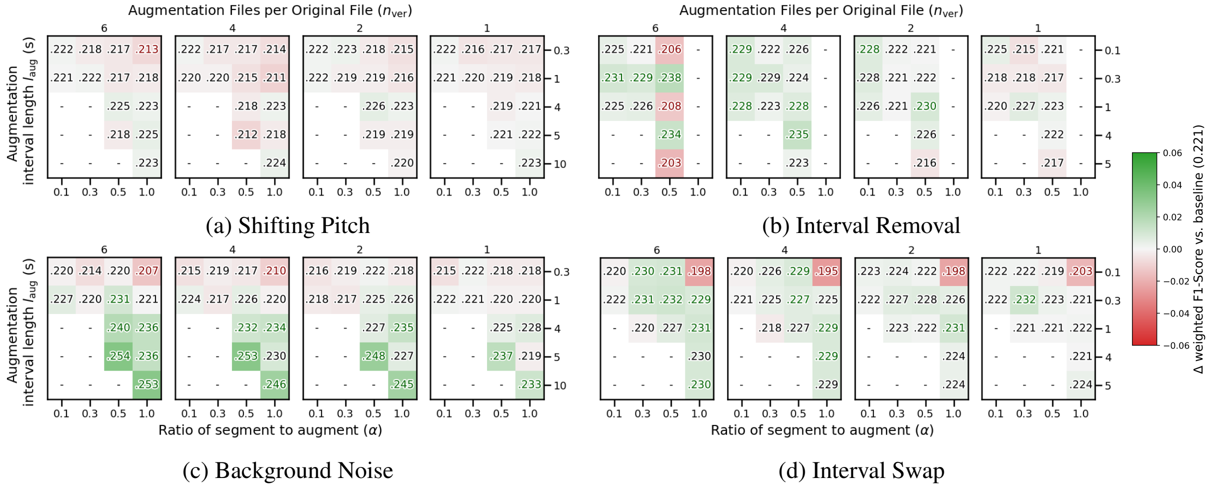
To span a range from local to full-segment changes, we use different sets of interval lengths depending on the augmentation method. Methods requiring an even number of intervals use lengths of $\{0.1, 0.3, 1, 4, 5\}$ seconds; other methods use $\{0.3, 1, 4, 5, 10\}$ seconds. Each original audio yields up to six augmented versions n_{ver} to remain computationally efficient. Augmentation is implemented in Python using Praat (Boersma and Weenink, 2021), Praat Vocal Toolkit (Corretge, 2012-2024) and Parselmouth (Jadoul et al., 2018).

2.4 Data Augmentation Techniques

The augmentation strategies in this work are inspired by the four basic operations of the EDA framework for text classification (Wei and Zou, 2019): Synonym Replacement, Random Deletion, Random Insertion, and Random Swap. Each forms the basis of one main audio-based augmentation technique, systematically tested across all previously described parameter combinations. Additional lightweight audio transformations are grouped according to the four main methods and evaluated using the best-performing settings (l_{aug}, α, n_{ver}) of their main category.

Synonym Replacement is realized through **Shifting Pitch (SP)**, which alters pitch while preserving timing and semantics. The target pitch is drawn from the typical male vocal range (80–170Hz) (Berg et al., 2017; Andreeva et al., 2014), with an added variation of ±10 Hz. Related methods include **Time Reversing (TR)**, which reverses the sample order; **Loudness Confusion (LC)**, which sets the peak amplitude of the interval to a random value in the range [0.2, 0.8]; **Time Stretching (TS)** (pitch-preserving) and **Speed Confusion**

[1] https://github.com/WoLFi22/DialectClassificationPipeline



(a) Shifting Pitch

(b) Interval Removal

(c) Background Noise

(d) Interval Swap

Figure 2: Mean weighted F1-scores for the four main augmentation methods across parameter settings. Cell colors indicate absolute deviation from the non-augmented baseline (colorbar right). Text color denotes significant improvement or decline relative to the baseline (green = better, red = worse, black = not significant), based on the two-sided Mann–Whitney U test with Holm–Bonferroni correction (see Section 2.2).

(SC), which modifies both pitch and speed via resampling. The last two use a factor in the range [0.8, 1.2], as supported by other works (Mushtaq et al., 2021; Salamon and Bello, 2017; Nanni et al., 2020; Ko et al., 2015). These methods preserve speaker identity and content but increase acoustic variation, analogous to lexical substitution in text.

Random Deletion is implemented as **Interval Removal (InR)**, which deletes intervals before resegmenting the remaining audio, similar to *SpliceOut* (Jain et al., 2022). A related approach is **Time Masking (TM)**, which zeroes out segments instead of deleting them. Both reduce local information density, akin to removing words in text.

Random Insertion is modeled by **Background Noise (BN)**, injecting noise from the MUSAN dataset (Snyder et al., 2015) with a random signal-to-noise ratio (SNR) between 0–30dB, reflecting a wide range of acoustic conditions. Additional techniques in this group include **Frequency Masking (FM)**, which applies band-stop filters to remove energy in 1-3 randomly chosen frequency bands between 100 Hz and 2500 Hz; **Frequency Swap (FS)**, where two frequency bands within the segment are swapped and **Frequency Insertion (FI)**, where selected frequency bands are replaced with corresponding bands extracted from another speaker of the same dialect class. FM, FS and FI are always applied over the entire segment. These methods introduce noise-like perturbations into the signal, analogous to random word insertions in text.

Random Swap is realized through **Interval Swap (InS)**, which exchanges two intervals of same length within a segment. **Speaker Insertion (SI)** follows a similar idea, replacing one interval with one from another speaker of the same class, as seen in related work (Xu et al., 2022). Like random word swaps in text, these methods alter sequence order while maintaining class identity.

3 Results

This section reports results for all evaluated augmentation methods. We first present the four main methods across parameter settings, followed by submethod analysis. The baseline (no augmentation) yields a median weighted F1 of **0.221** (±0.011), reflecting the difficulty of dialect classification and aligning with prior work (Stucki and Randjelovic, 2021; Jokisch and Dobbriner, 2019).

3.1 Main Augmentation Techniques

Figure 2 shows the results for the main methods. The best configuration per method is highlighted in the text and used for evaluating the corresponding submethods.

For **Shifting Pitch (SP)**, the best result is reached with l_{aug}*=4,* α*=0.5,* n_{ver}*=2*, yielding a non-significant 0.5% improvement over the baseline (see Figure 3). Overall, 4-second intervals perform best, whereas shorter and 5-second intervals tend to degrade performance, suggesting limited benefit of pitch shifting.

For **Interval Removal (InR)**, configurations with α=0.5, l_{aug}={0.1,1,5}, n_{ver}=6 degrade performance. This is likely due to identical segment start points. Introducing a small offset (0.25–1.5s in 0.25s steps), where each augmented version of a sample receives a different fixed offset before segmentation, significantly improves performance (adjusted $p < 0.0001$). The best setting, l_{aug}=*0.3, α=0.5, n_{ver}=6*, yields a 1.7% gain.

Background Noise (BN) performs best with one inserted noise interval per segment. The top configuration *(l_{aug}=5, α=0.5)* yields gains of 3.3% (n_{ver}=*6*), 3.2% (n_{ver}=4) and 2.7% (n_{ver}=2) over the baseline. Due to marginal differences between n_{ver}=4 and 6, the added cost of 6 may be unjustified. Short intervals (e.g. 0.3s) perform poorly, which may be due to shorter noise samples being used from the MUSAN dataset. Using only longer MUSAN samples of at least 5 seconds improves performance to 0.211, which remains below the baseline, though no longer statistically significant.

For **Interval Swap (InS)**, only l_{aug}=0.1, α=1.0 performs clearly worse, likely because the intervals are too short to capture meaningful linguistic content. This aligns with Zihlmann (2020), who report that vowels and consonants in Swiss Standard German typically last around 0.1s, with none exceeding 0.3s and dialectal vowels up to 0.25s. The best result is achieved with l_{aug}=0.3, α=0.5, n_{ver}=6, or alternatively at l_{aug}=*0.3, α=0.3, n_{ver}=1*, both yielding a 1.1% improvement. The choice of latter is particularly attractive, as it consistently improves performance across all tested file counts (n_{ver}=$\{1, 2, 4, 6\}$) while being computationally more efficient than the 6-file setup.

3.2 All Augmentation Techniques

LC and TR showed no effect on model performance, likely because DL-models ignore such low-level variations (Fischbach et al., 2025). TM and SI significantly worsened performance. For TM, the observed degradation appears linked to reduced information per segment: for a 10-second segment length, only 5.2 seconds remain unmasked. When increasing the segment length to 19.9 seconds to preserve 10 seconds of information, performance aligns with the baseline (adjusted p = 1.0). Applying an offset has no measurable effect, suggesting that TM, unlike InR, fails to introduce sufficient structural variation. SI, expected to reduce speaker bias, surprisingly performed poorly. Applying an offset mitigated the effect, resulting in a non-significant difference from the baseline (adjusted p = 1.0). This suggests that the initial drop was due to a lack of segment diversity, while SI itself introduces no beneficial variation.

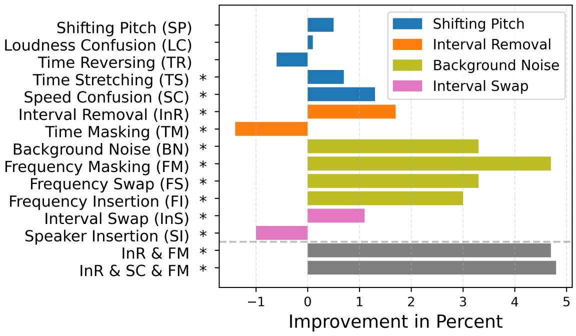


Figure 3: Relative improvement in weighted F1-score (%) over the baseline. Colors indicate augmentation groups; asterisks (*) mark statistically significant differences compared to baseline (two-sided Mann–Whitney U test with Holm–Bonferroni correction

All other methods performed significantly better than the baseline. TS and SC slightly outperformed SP, likely due to added temporal variation. The best results overall were achieved by FM, followed by FS, BN and FI. All belong to the same group, suggesting that frequency-based augmentation is highly effective for dialectal audio.

To reduce the computational cost, FM was combined with InR. This matched FM's performance while halving processing time due to fewer resulting augmented samples. Further combinations, such as with SC, offered only marginal (0.1%) improvements and were omitted for efficiency. Combining overlapping methods like FM+FS or SeS+SeR was avoided due to likely redundancy or fragmentation.

4 Conclusion

The best result was achieved using Frequency Masking, yielding a 4.7% improvement in weighted F1-score over the unaugmented baseline. When combined with Interval Removal (InR), the performance remained the same, while computational effort was significantly reduced by halving the number of augmented training samples. Although InR on its own did not improve performance, its efficiency benefits make it a valuable addition in combination. Future work should explore fine-grained tuning of submethod hyperparameters and investigate whether increasing the number of augmented files per original leads to saturation or further gains.

Acknowledgments

This research is supported by the Academy of Science and Literature Mainz (grant REDE 0404), the German Federal Ministry of Education and Research (BMBF) (grant AnDy 16DKWN007), the state of North-Rhine Westphalia as part of the Lamarr-Institute for Machine Learning and Artificial Intelligence, LAMARR22B and the Research Center Deutscher Sprachatlas Marburg.

References

Bistra Andreeva, Grazyna Demenko, Magdalena Wolska, Bernd Möbius, Frank Zimmerer, Jeanin Jügler, Magdalena Oleskowicz-Popiel, and Jürgen Trouvain. 2014. Comparison of pitch range and pitch variation in slavic and germanic languages. In *Proceedings to the 7th Speech Prosody Conference. Trinity College Dublin, Ireland. May 20-23, 2014*, pages 776–780. International Speech Communication Association.

Martin Berg, Michael Fuchs, Kerstin Wirkner, Markus Loeffler, Christoph Engel, and Thomas Berger. 2017. The speaking voice in the general population: normative data and associations to sociodemographic and lifestyle factors. *Journal of Voice*, 31(2):257–e13.

Paul Boersma and David Weenink. 2021. Praat: doing phonetics by computer [Computer program]. Version 6.1.38, retrieved 2 January 2021 `http://www.praat.org/`.

Stefan Braun, Daniel Neil, and Shih-Chii Liu. 2017. A curriculum learning method for improved noise robustness in automatic speech recognition. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 548–552. IEEE.

Ramon Corretge. 2012-2024. Praat Vocal Toolkit. retrieved 20 January 2024 `https://www.praatvocaltoolkit.com`.

Lucas Ferreira-Paiva, Elizabeth Alfaro-Espinoza, Vinicius M Almeida, Leonardo B Felix, and Rodolpho VA Neves. 2022. A survey of data augmentation for audio classification. In *Congresso Brasileiro de Automática-CBA*, volume 3.

Lea Fischbach, Caroline Kleen, Lucie Flek, and Alfred Lameli. 2025. Does preprocessing matter? An analysis of acoustic feature importance in deep learning for dialect classification. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, pages 159–169, Tallinn, Estonia. University of Tartu Library.

Takashi Fukuda, Raul Fernandez, Andrew Rosenberg, Samuel Thomas, Bhuvana Ramabhadran, Alexander Sorin, and Gakuto Kurata. 2018. Data augmentation improves recognition of foreign accented speech. In *Interspeech*, September, pages 2409–2413.

Sture Holm. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6:65–70.

Yannick Jadoul, Bill Thompson, and Bart de Boer. 2018. Introducing Parselmouth: A Python interface to Praat. *Journal of Phonetics*, 71:1–15.

Arjit Jain, Pranay Samala, Deepak Mittal, Preethi Jyothi, and Maneesh Singh. 2022. Spliceout: A simple and efficient audio augmentation method. In *Interspeech*, pages 2678–2682.

Oliver Jokisch and Johanna Dobbriner. 2019. Text-independent dialect classification in read and spontaneous speech. In *Proceedings of the 1st International Conference on Language Technologies for All*, pages 350–354, Paris, France. European Language Resources Association (ELRA).

Gwantae Kim, David K. Han, and Hanseok Ko. 2021. Specmix: a mixed sample data augmentation method for training with time-frequency domain features. In *22nd Annual Conference of the International Speech Communication Association, INTERSPEECH 2021*, Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, pages 6–10. International Speech Communication Association.

Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. 2015. Audio augmentation for speech recognition. In *Interspeech*, volume 2015, page 3586.

H. B. Mann and D. R. Whitney. 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, 18(1):50 – 60.

Zohaib Mushtaq and Shun-Feng Su. 2020. Environmental sound classification using a regularized deep convolutional neural network with data augmentation. *Applied Acoustics*, 167:107389.

Zohaib Mushtaq, Shun-Feng Su, and Quoc-Viet Tran. 2021. Spectral images based environmental sound classification using cnn with meaningful data augmentation. *Applied Acoustics*, 172:107581.

Loris Nanni, Gianluca Maguolo, and Michelangelo Paci. 2020. Data augmentation approaches for improving animal audio classification. *Ecological Informatics*, 57:101084.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *Interspeech 2019*.

Justin Salamon and Juan Pablo Bello. 2017. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal processing letters*, 24(3):279–283.

Jürgen Erich Schmidt, Joachim Herrgen, Roland Kehrein, and Alfred Lameli. 2020ff. Regionalsprache.de. Forschungsplattform zu den modernen Regionalsprachen des Deutschen. Forschungszentrum Deutscher Sprachatlas Marburg.

Skipper Seabold and Josef Perktold. 2010. statsmodels: Econometric and statistical modeling with python. In *9th Python in Science Conference*.

Joel Shor and Subhashini Venugopalan. 2022. TRILLsson: Distilled Universal Paralinguistic Speech Representations. In *Proc. Interspeech 2022*, pages 356–360.

David Snyder, Guoguo Chen, and Daniel Povey. 2015. MUSAN: A Music, Speech, and Noise Corpus. ArXiv:1510.08484v1.

Samuel Stucki and Patrik Randjelovic. 2021. Automatic detection of swiss german dialects using wav2vec. Project Thesis, ZHAW School of Engineering, Centre for Artificial Intelligence. Accessed: July 17, 2025.

Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Peter Wiesinger. 1983. Die Einteilung der deutschen Dialekte. In Werner Besch, editor, *Dialektologie: Ein Handbuch zur deutschen und allgemeinen Dialektforschung*, volume 1.2 of *Handbücher zur Sprach- und Kommunikationswissenschaft*, pages 807–900. Berlin/New York: de Gruyter, Berlin, New York.

Xin-Shun Xu, Zhuangzhi Chen, Dongwei Xu, Huaji Zhou, Shanqing Yu, Shilian Zheng, Qi Xuan, and Xiaoniu Yang. 2022. Mixing signals: Data augmentation approach for deep learning based modulation recognition. *ArXiv*, abs/2204.03737.

Urban Zihlmann. 2020. Vowel and consonant length in four alemannic dialects and their influence on the respective varieties of swiss standard german. *Wiener Linguistische Gazette*, 86:1–46.