Interspeech 2024
1-5 September 2024, Kos, Greece

Privacy PORCUPINE: Anonymization of Speaker Attributes Using Occurrence Normalization for Space-Filling Vector Quantization

Mohammad Hassan Vali, Tom Bäckström

Aalto University, Department of Information and Communications Engineering, Finland
mohammad.vali@aalto.fi, tom.backstrom@aalto.fi

Abstract

Speech signals contain a vast range of private information such as its text, speaker identity, emotions, and state of health. Privacy-preserving speech processing seeks to filter out any private information that is not needed for downstream tasks, for example with an information bottleneck, sufficiently tight that only the desired information can pass through. We however demonstrate that the occurrence frequency of codebook elements in bottlenecks using vector quantization have an uneven information rate, threatening privacy. We thus propose to use space-filling vector quantization (SFVQ) together with occurrence normalization, balancing the information rate and thus protecting privacy. Our experiments with speaker identification validate the proposed method. This approach thus provides a generic tool for quantizing information bottlenecks in any speech applications such that their privacy disclosure is predictable and quantifiable.

Index Terms: anonymization, differential privacy, privacy-preservation, space-filling curves, vector quantization

1. Introduction

Speech is a convenient medium for interacting between humans and with technology, yet evidence demonstrates that it exposes speakers to threats to their privacy. A central issue is that, besides the linguistic content which may be private, speech contains also private side information such as the speaker's identity, age, state of health, ethnic background, gender identity, and emotions. Revealing such sensitive information to a listener may expose the speaker to threats such as price gouging, tracking, harassment, extortion, and algorithmic stereotyping. To protect speakers, privacy-preserving speech processing seeks to anonymize speech signals by stripping away private information that is not required for the downstream task [1, 2].

A common operating principle for privacy-preserving speech processing is to pass information through an information bottleneck that is tight enough to allow through only the desired information and prevent transmission of any other private information [3]. Such bottlenecks can be implemented for example as *autoencoders*, where a neural network, known as the encoder, compresses information to a bottleneck, and a second network, the decoder, reconstructs the desired output [4]. The information rate of the bottleneck can be quantified absolutely, only if it is quantized, and quantization is thus a required component of any proof of privacy [2]. Quantization of the bottleneck can be implemented either in scalar or vector form, but vector quantization is preferred because it is easier to implement such that it operates near the optimal limit of rate-distortion [5].

As pointed out in the Privacy ZEBRA framework [1], we need to characterize privacy protections both in terms of average disclosure of private information (in bits) as well as worst-case disclosure. Vector quantization (VQ) is a constant-bitrate quantizer in the sense that it quantizes the input with a codebook of K elements, and the indices of such a codebook can be expressed with $B = \log_2 K$ bits. This corresponds to the average disclosure of private information. In terms of the worst-case, it is well-known that different codebook elements are used with different frequencies (see fig. 1a) and this information can be used for optimal coding [7, 5]. This means that a relatively smaller subset of speakers could be assigned to a particular codebook index, such that any time a speaker is assigned to that codebook index, the range of possible speakers is relatively smaller than for other codebook indices and the corresponding disclosure of private information is larger. However, to the best of the authors' knowledge, this disclosure has not previously been quantified nor do we have prior solutions for compensating for such an increase in disclosure. In this paper we only care about improving worst-case disclosure. We also define only one embedding per speaker and the trivial solution that all speakers would be assigned to a particular codebook index is discarded.

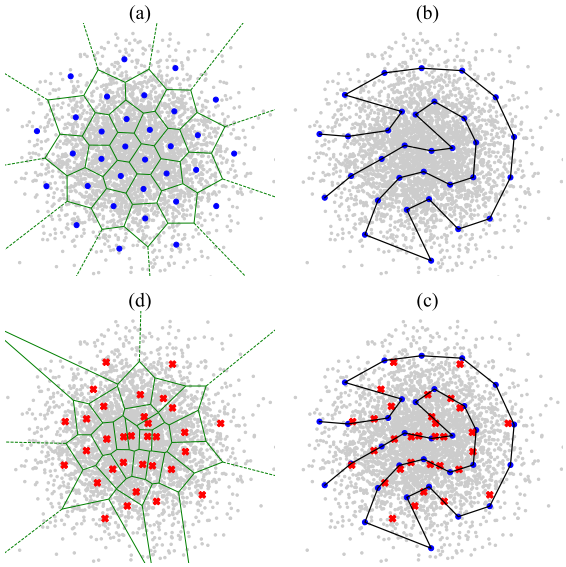

Figure 1: *(a) Codebook vectors (blue points) of a 5 bit vector quantization applied on a Gaussian distribution (gray points); Voronoi-regions are shown in green. (b) Space-filling vector quantization (SFVQ) [6] with 5 bit applied on the same Gaussian distribution (curve in black with codebook vectors in blue). (c) Same SFVQ curve together with the resampled codebook vectors (red crosses). (d) Same resampled codebook vectors along with their Voronoi-regions (in green).*

10.21437/Interspeech.2024-117

As a solution, we use here our recently proposed modification of VQ known as space-filling vector quantization (SFVQ) [6], which incorporates space-filling curves into VQ. Space-filling curves refer to piece-wise continuous lines defined by recursive rules, such that when the recursion approaches infinity without bound, the curve will fill an N-dimensional space with $N > 1$. In our case, we define the SFVQ as the linear continuation of a vector quantizer, such that subsequent codebook elements are connected with a line where inputs can be mapped to any point on that piece-wise continuous line (see fig. 1b). To avoid diverging codebooks during training, we can use dithering such that the codebook for a particular batch consists of randomly shifted points on the line. Furthermore, to train the SFVQ in an end-to-end training of neural networks, we can use noise substitution in vector quantization (NSVQ) [8] technique during training.

This paper proposes to use occurrence normalization for SFVQ [6], named Privacy PORCUPINE, where a vector quantizer is resampled along the SFVQ's curve, such that all elements of the resampled codebook have equal occurrence likelihood (see figs. 1c and 1d). We demonstrate in theory and with experiments, that, in the worst-case, VQ can disclose private information at an infinite bitrate, while the Privacy PORCUPINE can potentially reach the theoretical limit where worst-case disclosure is equal to the maximum average disclosure. The implementation of our proposed method is publicly available at https://github.com/Speech-Interaction-Technology-Aalto-U/Privay-PORCUPINE.git

2. Space-filling vector quantization

Conventional vector quantization models data by assigning inputs x to the best matching codebook entry c_k, defined as

$$\begin{aligned} k^* &= \arg\min_k \|x - c_k\|^2 \\ \hat{x}_{VQ} &= c_{k^*}, \end{aligned} \tag{1}$$

where $\|\cdot\|^2$ refers to Euclidean distance and $\hat{x}_{VQ}$ is the output of the vector quantizer. Figure 1a illustrates the Voronoi-regions of a vector quantizer where all inputs are assigned to a particular codebook entry.

Space-filling vector quantization [6] is a linear continuation of vector quantization, where inputs are mapped to the line connecting subsequent codebook entries

$$\begin{aligned} k^*, \lambda^* &= \arg\min_{k,\lambda} \|x - ((1-\lambda)c_k + \lambda c_{k+1})\|^2 \\ \hat{x}_{SFVQ} &= (1-\lambda^*)c_{k^*} + \lambda^* c_{k^*+1}, \end{aligned} \tag{2}$$

where λ is the interpolation coefficient between subsequent codebook vectors. Finding the global optimum for λ is computationally expensive, and thus during training, we use a random λ for each batch (sampled from the uniform distribution of $U(0,1)$) and optimize only the indices k. The curve obtained is illustrated fig. 1b.

During inference, we first find the optimal index k^*, similar to classical vector quantization using eq. (1). Keeping k fixed, we can then solve the optimal λ by setting the derivative of the norm in eq. (2) to zero and obtain

$$\lambda^* = \frac{(c_k - c_{k+1})^T (c_k - x)}{\|c_k - c_{k+1}\|^2}, \tag{3}$$

but limited to $\lambda \in [0,1]$. The interpolation coefficient λ has to be calculated for both of the intervals $k-1, k$ and $k, k+1$, and out of those, we choose the λ corresponding to the interval that minimizes the error of $\|x - \hat{x}_{SFVQ}\|^2$.

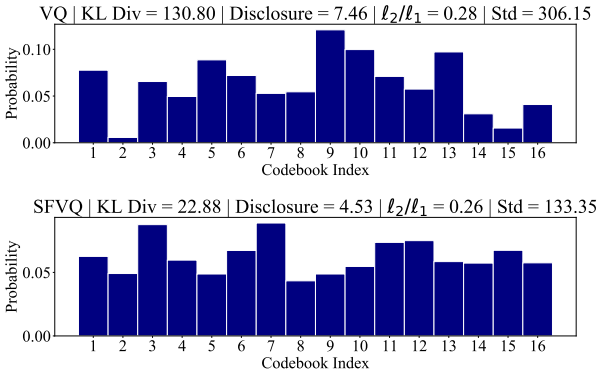

Figure 2: *Histogram of codebook indices frequencies for (top) vector quantization and (bottom) resampled space-filling vector quantization in case of 4 bit quantization; KL Div refer to Kullback-Leibler divergence of the histograms to the expected binomial distribution $f(k)$ (discussed in section 6). Disclosure, $\frac{\ell_2}{\ell_1}$ and Std are the worst-case disclosure, sparseness and standard deviation of the histograms.*

3. Occurrence frequency in quantization

Figure 1a demonstrates that in areas where inputs are less likely, the areas quantized to a particular codebook entry, known as Voronoi-regions, are larger. Though such Voronoi-regions are larger, they still contain a smaller number of input samples. Similarly, small Voronoi-regions have a larger number of input samples. Such differences are due to the optimization of the codebook by minimizing the mean square error criteria; more common inputs are quantized with a smaller error to minimize the average error [9]. Our objective is to determine how such uneven distribution of inputs to codebook entries influences the disclosure of private information.

We measure disclosure in terms of how much the population size of possible identities for an unknown speaker is decreased with a new observation. In other words, assume that we have prior knowledge that the speaker belongs to a population of size M. If we have an observation that the speaker is quantized to an index k, we have to evaluate how many speakers L out of M will be quantized to the same index. This decrease can be quantified by the ratio of populations $\frac{L}{M}$ corresponding to the disclosure of $B_{\text{leak}} = \log_2 \frac{M}{L}$ bits of information.

At the extreme, it is possible that only a single speaker may be quantized to a particular index. This means that only one speaker $L = 1$ is quantized to the bin out of an arbitrarily large M, though in practice we can verify results only for finite M. Still, in theory, if $M \to \infty$, then also the disclosure diverges $B_{\text{leak}} \to \infty$ bits. The main objective of this paper is to modify vector quantization to prevent such catastrophic leaks.

4. Codebook resampling

After training the space-filling vector quantizer (SFVQ) [6] on the training set, comprising embeddings for a population of M speakers (fig. 1b), we map all the M embedding vectors onto the learned curve. To normalize the occurrence frequency using K codebook vectors, each codebook element has to represent M/K speaker embeddings. In other words, each Voronoi-region should encompass M/K speaker embeddings. Considering these M mapped embeddings on SFVQ's curve, we start from the first codebook element (one end of the curve), take the

first M/K mapped embeddings, and calculate the average of these vectors. We define this average vector as the new resampled codebook vector (red crosses in fig. 1c) representing the first chunk of M/K speaker embeddings. Then similarly, we continue until we compute all K resampled codebook vectors for all K chunks of M/K speaker embeddings. As SFVQ maps the input data batch to the curve (i.e. the lines connecting subsequent codebook vectors) during training, the space-filling lines are mainly located inside the data distribution [6]. Hence, when we calculate the resampled codebook vector as the average of M/K mapped embeddings on SFVQ's curve, the resampled vector will reside inside the data distribution (see fig. 1d).

5. Experiments

To gain access to a large number of speakers, we selected the Common Voice corpus (version 16.1) [10]. Skipping audio clips shorter than 2 and over 60 seconds, the dataset includes 2 318 715 unique audio clips from 90 224 unique speakers. We selected 10 240 speakers randomly as the test set such that the train set has 79 984 speakers. To compute speaker embeddings, we used the pretrained speaker verification model of ECAPA-TDNN [11] using SpeechBrain [12]. Since the pretrained ECAPA-TDNN model was not trained on the Common Voice dataset, we did an initial validation test to demonstrate that this pretrained model gives acceptable speaker embeddings on this dataset. We randomly selected 10 000 unique speakers and for each of them, we took two audio clips from the same speaker and one audio file from a different speaker. Then, for each speaker we did two speaker verification tests; one over two audio files from the same speakers and the other over two audio files from different speakers. This gave a speaker verification accuracy of 96.46 %, which validates the use of the ECAPA-TDNN model with the Common Voice dataset.

To obtain the train and test set vectors, we assigned only one embedding vector to each individual speaker, such that for the speakers that have more than one audio clip, we averaged over embeddings extracted from all available audio clips (up to 100 clips) from that specific speaker. We trained vector quantization (VQ) and space-filling vector quantization (SFVQ) methods on the train set for 100 epochs with a batch size of 64. We used the Adam optimizer with the initial learning rate of 10^{-2} which is halved after 60 and 80 training epochs. For more comprehensive experimentation, we trained both methods for different bitrates ranging from 4 bit to 10 bit (16 to 1024 codebook vectors). For each individual quantization bitrate, we repeated the experiments 20 times to enable statistical analysis of confidence of the evaluation metrics.

6. Evaluation metrics

Suppose we have K codebook vectors (bins) to quantize a population of M speakers. Our target is to achieve a uniform distribution of samples onto the K codebook vectors, $U(1, K)$, such that every codebook vector is used $\frac{M}{K}$ times (see fig. 2). If we sample M integers from the uniform distribution of $U(1, K)$, we will obtain the histogram $h(k)$. Then, if we take the histogram of occurrences in the bins of $h(k)$ (i.e. histogram of histogram), we will see that the new histogram follows a binomial distribution $f(k)$ such that

$$f(k) = \Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}, \qquad (4)$$

where the random variable X is the occurrence in each bin, n is the number of trials ($n = M$), and p is the success probability for each trial ($p = 1/K$). After obtaining the histogram of codebook indices occurrences $g(k)$ (fig. 2) for both VQ and resampled SFVQ methods, we compute the histogram of occurrences in the bins of $g(k)$ denoted by $\hat{f}(k)$. The binomial distribution $f(k)$ is the theoretical optimum, to which our observation $\hat{f}(k)$ should coincide. Hence, we use Kullback-Leibler (KL) divergence between $f(k)$ and $\hat{f}(k)$ to assess the distance between the observed and the ideal distributions.

By having the histogram of occurrences $g(k)$, we calculate the minimum of the histogram divided by the total number of samples ($M = \sum_k g(k)$) as the worst-case disclosure. We also compute the average disclosure as entropy of occurrence frequencies $g(k)/M$. In addition, we use a sparseness measure (ℓ_2/ℓ_1) [13] and standard deviation (Std) as heuristic measures of uniformity of the obtained histogram $g(k)$. For all evaluation metrics except average disclosure, a lower value means that the histogram of codebook occurrence $g(k)$ is closer to uniform distribution (ideal case). The metrics are defined as

$$\begin{aligned}
KL_{Div} &= \textstyle\sum_k \hat{f}(k) \log_2 \left(\frac{\hat{f}(k)}{f(k)} \right), \\
\text{Worst-case disclosure} &= -\log_2 \frac{\min g(k)}{M}, \\
\text{Average disclosure} &= -\textstyle\sum_k \frac{g(k)}{M} \log_2 \frac{g(k)}{M}, \\
\text{Sparseness} &= \frac{\ell_2}{\ell_1} = \frac{\sqrt{\sum_k g^2(k)}}{\sum_k |g(k)|}, \\
\text{Std} &= \sqrt{\tfrac{1}{M} \textstyle\sum_k \left(g(k) - \frac{M}{K} \right)^2}.
\end{aligned}$$

7. Results and discussions

Figure 2 illustrates the occurrence frequencies for vector quantization (VQ) and the proposed space-filling vector quantizer with occurrence normalization, both with 4 bit of information corresponding to 16 codebook vectors. By informal visual inspection we can see that entries in the proposed method are more uniformly distributed, as desired, but to confirm results we need a proper evaluation.

Figure 3 illustrates the performance of both methods as a function of bitrate. In each case and for all bitrates, the proposed method (blue line) is below vector quantization (red line), indicating that the leakage of private information is smaller for the proposed method. With the KL-divergence (fig. 3a), the difference is clear and there is no overlap between confidence intervals. For worst-case disclosure (fig. 3b), the difference is clearly larger for lower bitrates while we have some overlap for confidence intervals at high bitrates. The proposed method makes the average disclosure to be slightly higher than VQ, whereas the average disclosure for both proposed method and VQ are extremely close to the upper bound of average disclosure where the histogram of occurrence frequencies is perfectly flat. The ℓ_2/ℓ_1 sparseness measure does show a statistically significant difference throughout all bitrates, though the relative difference is small. The standard deviation (fig. 3d) has also no overlap, showing a statistically significant difference through all bitrates.

All measures (except worst-case disclosure) improve with increasing bitrate, indicating that privacy leaks are reduced with the increase in model's bitrate. Importantly, the reduction in KL-divergence (in bits) is larger than the increase in bitrate (also in bits) and the reduction applies to both methods. The reduction in divergence is thus not merely mirroring the increase in

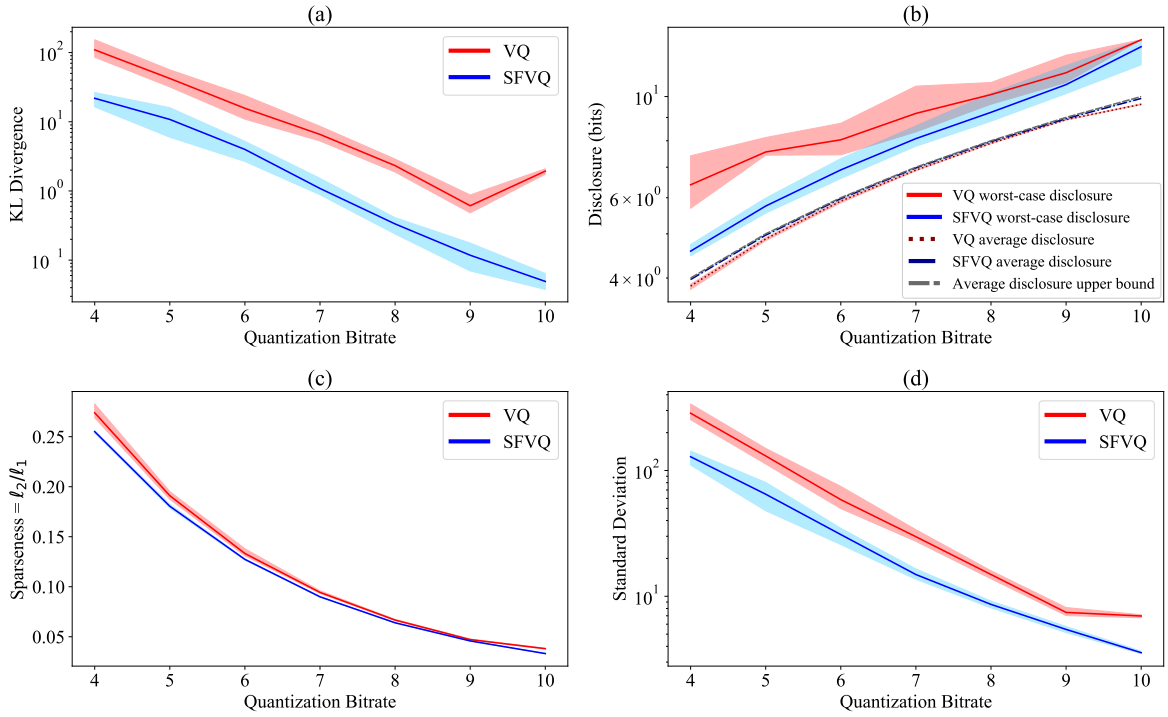

Figure 3: *Performance of vector quantization (VQ) and the proposed space-filling vector quantizer (SFVQ) with occurrence normalization, as a function of bitrate, in terms of (a) KL-divergence, (b) disclosure, (c) sparseness, and (d) standard deviation. The solid lines refer to the mean value of the metrics over 20 repetitions, and corresponding filled areas refer to their 95 % quantiles.*

bitrate but suggests that both quantizers are more evenly covering the source distribution when the bitrate is increased.

The difference in KL-divergence between the two methods is roughly uniform on the logarithmic scale over all bitrates. This indicates that the ratio of improvement gained by using occurrence normalization is constant, roughly $\frac{KL_{DIV}(VQ)}{KL_{DIV}(SFVQ)} \approx 5$, over all bitrates. The only exception is that vector quantization, when moving from 9 bit to 10 bit, shows a curious increase in all measures, corresponding to a degradation in performance. We hypothesize that this is a critical threshold beyond which outliers begin to become visible in vector quantization as isolated individuals. We have not observed any indications of similar weaknesses with the proposed method.

Another interesting feature in fig. 3a is that the logarithm of KL-divergence is in a practically linear relationship with the bitrate, from a KL-divergence of approximately 11 at 4 bit to approximately 0.03 at 10 bit.

When comparing the different performance measures, the KL-divergence thus behaves in the most consistent way. It also has the benefit that it expresses the difference in the optimal distribution in terms of bits. The KL-divergence can thus approximate the leakage of private information in bits, offering an intuitively understandable domain for analysis.

8. Conclusion

Privacy-preserving speech processing is becoming increasingly important as the usage of speech technology is increasing. By removing superfluous private information from a speech signal by passing it through a quantized information bottleneck, we can gain provable protections for privacy. Such protections however rely on the assumption that quantization levels are used with equal frequency. Our theoretical analysis and experiments demonstrate that vector quantization, optimized with the minimum mean square criterion, does generally not provide such uniform frequencies. In the worst case, some speakers could be uniquely identified even if the quantizer on average provides ample protection.

We propose an occurrence normalization approach to avoid such privacy threats. It is based on resampling the codebook vectors using space-filling vector quantization such that each entry is accessed with equal frequency. The protection of privacy is thus achieved by increasing the quantization error for inputs that occur less frequently, while more common inputs gain better accuracy. This is in line with the theory of differential privacy [14].

Our experiments with speaker identification confirm that occurrence normalization indeed improves the spread over all quantization levels and provides a roughly 5-fold improvement in the KL-divergence. The improvement in privacy protection is remarkably consistent over all bitrates tested.

We used speaker identification as an illustrative application, though the proposed approach can be used in gaining a provable reduction of private information of any attributes of speech. Such quantization can take at least two principal approaches (a) we can quantize an attribute such as phonemes, *to eliminate any other attributes* from passing through a bottleneck [6], or (b) we can quantize attributes like the age or identity of a speaker, *to reduce accuracy*, to retain rough characteristics of the speech signal, but anonymize the specific identity.

In summary, occurrence normalization for space-filling vector quantization is a generic tool for privacy-preserving speech processing. It provides a method for quantifying the amount of information passing through an information bottleneck and thus forms the basis of speech processing methods with provable privacy.

9. References

[1] A. Nautsch, J. Patino, N. Tomashenko, J. Yamagishi, P.-G. Noé, J.-F. Bonastre, M. Todisco, and N. Evans, "The Privacy ZEBRA: Zero Evidence Biometric Recognition Assessment," in *Proc. Interspeech 2020*, 2020, pp. 1698–1702. [Online]. Available: https://doi.org/10.21437/Interspeech.2020-1815

[2] T. Bäckström, "Privacy in speech technology," submitted to *Proc IEEE*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2305.05227

[3] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000. [Online]. Available: https://arxiv.org/abs/physics/0004057

[4] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with VQ-VAE-2," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/5f8e2fa1718d1bbcadf1cd9c7a54fb8c-Paper.pdf

[5] T. Bäckström, J. Lecomte, G. Fuchs, S. Disch, and C. Uhle, *Speech coding: with code-excited linear prediction*. Springer, 2017. [Online]. Available: https://doi.org/10.1007/978-3-319-50204-5

[6] M. H. Vali and T. Bäckström, "Interpretable latent space using space-filling curves for phonetic analysis in voice conversion," in *Proc. Interspeech*, 2023. [Online]. Available: https://doi.org/10.21437/Interspeech.2023-1549

[7] J. Rissanen and G. G. Langdon, "Arithmetic coding," *IBM Journal of research and development*, vol. 23, no. 2, pp. 149–162, 1979. [Online]. Available: https://doi.org/10.1147/rd.232.0149

[8] M. H. Vali and T. Bäckström, "NSVQ: Noise substitution in vector quantization for machine learning," *IEEE Access*, 2022. [Online]. Available: https://doi.org/10.1109/ACCESS.2022.3147670

[9] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Springer Science & Business Media, 1991, vol. 159. [Online]. Available: https://doi.org/10.1007/978-1-4615-3626-0

[10] "Mozilla common voice corpus," 2024, accesed on June 11, 2024. [Online]. Available: https://commonvoice.mozilla.org/en/datasets

[11] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Proc. Interspeech*, 2020, pp. 3830–3834. [Online]. Available: https://doi.org/10.21437/Interspeech.2020-2650

[12] M. Ravanelli, T. Parcollet, P. Plantinga, A. Rouhe, S. Cornell, L. Lugosch, C. Subakan, N. Dawalatabad, A. Heba, J. Zhong, J.-C. Chou, S.-L. Yeh, S.-W. Fu, C.-F. Liao, E. Rastorgueva, F. Grondin, W. Aris, H. Na, Y. Gao, R. D. Mori, and Y. Bengio, "SpeechBrain: A general-purpose speech toolkit," 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2106.04624

[13] N. Hurley and S. Rickard, "Comparing measures of sparsity," *IEEE Transactions on Information Theory*, vol. 55, no. 10, pp. 4723–4741, 2009. [Online]. Available: https://doi.org/10.1109/TIT.2009.2027527

[14] C. Dwork, "Differential privacy: A survey of results," in *International conference on theory and applications of models of computation*. Springer, 2008, pp. 1–19. [Online]. Available: https://doi.org/10.1007/978-3-540-79228-4_1