AUDIO SUPER-RESOLUTION USING CONCATENATIVE RESYNTHESIS

Michael I Mandel

Brooklyn College, CUNY
Computer & Information Science
Brooklyn, NY 11210
mim@sci.brooklyn.cuny.edu

Young Suk Cho

The Ohio State University
Computer Science & Engineering
Columbus, OH, 43210
cho.648@osu.edu

ABSTRACT

This paper utilizes a recently introduced non-linear dictionary-based denoising system in another voice mapping task, that of transforming low-bandwidth, low-bitrate speech into high-bandwidth, high-quality speech. The system uses a deep neural network as a learned non-linear comparison function to drive unit selection in a concatenative synthesizer based on clean recordings. This neural network is trained to predict whether a given clean audio segment from the dictionary could be transformed into a given segment of the degraded observation. Speaker-dependent experiments on the small-vocabulary CHiME2-GRID corpus show that this model is able to resynthesize high quality clean speech from degraded observations. Preliminary listening tests show that the system is able to improve subjective speech quality evaluations by up to 50 percentage points, while a similar system based on non-negative matrix factorization and trained on the same data produces no significant improvement.

Index Terms— bandwidth expansion; concatenative synthesis; deep neural networks; nonparametric; speech

1. INTRODUCTION

While frequencies between 300 and 3400 Hz are sufficient for reasonable intelligibility of speech over telephone lines, such speech has much lower quality than wider-bandwidth alternatives [1]. Coding algorithms used in mobile telephony can further reduce the bitrates necessary to transmit intelligible speech, at the cost of further quality reductions [2], especially when signals are decoded and re-encoded to cross between carriers' networks. Furthermore, both intelligibility and quality suffer when packets are lost between sender and receiver [3, 4]. This paper presents a method that can utilize the growing memory capacity of mobile devices to recover the quality of the original speech after such low-bitrate transmission. Mandel et al [5] introduced the concatenative resynthesis system, which is able to transform noisy speech into clean speech using a concatenative synthesizer and a learned, non-linear function to match clean dictionary "chunks" of speech (200 ms segments) to noisy observed chunks. Because of the flexibility of this learning process, it can be used for many other mappings between different types of speech. In the case of the current paper, the system learns to map from speech that has gone through reverberation, coding, and packet loss, back to the original wideband anechoic speech.

This work was supported by a Google Faculty Research Award

2. RELATED WORK

A number of methods exist for expanding the bandwidth of a signal from narrowband (4 kHz nyquist) to wideband (8 kHz nyquist) or beyond [6]. These methods typically operate on a single short frame of audio at a time, perhaps 20–30 ms, attempting to predict higher frequencies' envelope [7, 8] and excitation [9, 10] signals from those of the lower frequencies. Such approaches generally do not consider longer-term contextual information that can be helpful in deciphering ambiguous inputs and in creating continuity over time. More general than bandwidth expansion is the problem of spectral imputation, which treats observed spectral regions as reliable and attempts to estimate unobserved spectral regions from them [11–14]. None of these methods can remove artifacts from coding or packet loss, as they assume that the low bandwidth information is reliable. A similar system with the same constraint is that of [15], which uses approximate matching pursuit to impute missing spectral regions using an overcomplete dictionary. In our experiments in Section 4, we compare our approach to that of [11], which performs bandwidth expansion using non-negative matrix factorization separately on excitation and envelope spectra. Such approaches can be trained more efficiently than ours by using mixtures of multiple sources, but are hindered in their recovery from non-linear distortions by their assumption of linear mixing.

3. NON-LINEAR DICTIONARY-BASED BANDWIDTH ENHANCEMENT

Figure 1 shows a schematic of the deep neural network affinity function used in the concatenative resynthesis system. While standard deep neural networks used for speech processing take as input a single "chunk" of the input signal (a target frame and several surrounding context frames), ours instead takes a pair of chunks, one from the input signal and one from the dictionary. It then outputs a single value, an estimate of the affinity between the two chunks, where affinity is defined as the probability that the dictionary chunk could be transformed into the observed input chunk. This probability is learned by example from training data, so the system should be able to learn transformations where parallel source (damaged) and target (clean) signals are available. This network is used to compute the affinity between all pairs of clean chunks in the dictionary and noisy chunks from the observation, forming a similarity matrix. The best sequence of dictionary chunks can be found using the Viterbi algorithm, which combines this similarity matrix with a transition affinity matrix between dictionary chunks. The original audio corresponding to each chunk in this optimal sequence is then re-assembled using overlap-add into the resynthesized clean audio. Because the

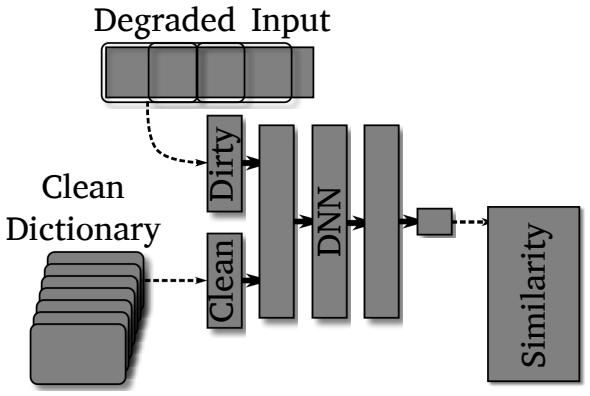


Figure 1: Overview of the concatenative resynthesis affinity function. Each "chunk" of the observed degraded signal is paired with each "chunk" of clean speech in the dictionary as input to a deep neural network (DNN), which outputs an estimation of the probability that the clean chunk was transformed into the degraded chunk.

dictionary only consists of high-quality speech, this resynthesis contains no noise aside from incorrect or misplaced dictionary elements. Mathematically, the best path is

$$\hat{\mathbf{z}} = \underset{\mathbf{z}}{\operatorname{argmax}} \prod_t p(z_t = j \mid x_t) p(z_t = j \mid z_{t-1} = i) \quad (1)$$

$$= \underset{\mathbf{z}}{\operatorname{argmax}} \prod_i g(z_j, x_i) T_{ij} \quad (2)$$

where $g(z_j, x_i)$ is the neural network affinity function between clean chunk z_j and degraded chunk x_i defined in Section 3.1, T_{ij} is the transition matrix defined in Section 3.2, and the maximization is over all possible paths, $\mathbf{z}$.

3.1. Learning the similarity function

We thus endeavor to learn a non-linear affinity function, $g(z, x)$ between a clean speech chunk, z_j, and a degraded speech chunk, x_i. We are using chunks that are 192 ms long, corresponding to 11 frames of length 32 ms offset by 16 ms from each other. Both the clean and distorted input chunk are transformed into the log mel spectrum [16], but any linear or non-linear features could be used. To train the network, we use parallel clean and distorted signals generated for this purpose, as illustrated in Figure 2. In particular, clean speech is convolved with a room impulse response, coded using a mobile telephony voice codec, and optionally passed through a packet loss simulator. Because we have generated the distorted data in parallel with the clean, we assign the desired output of the network, y_{ij}, to be 1 if the degraded chunk was generated from the clean chunk, and 0 otherwise. We use a balanced set of positive and negative pairs to train the network, where each degraded chunk is involved in two pairs, one with its corresponding clean chunk, and one with a mis-matched clean chunk. See Figure 2 for an illustration of how this ground truth target data is derived from clean and degraded speech.

In this work, we use a deep neural network (DNN) as $g(z, x)$. It takes as input 11 frames of the 22-channel log mel spectrograms of a clean and a noisy chunk, rasterized and concatenated together, making a total of 484 dimensions. It has 4 hidden layers of 1024

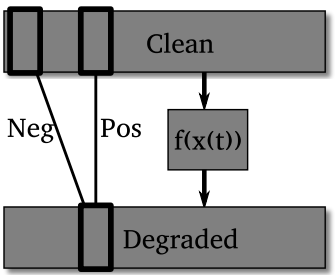


Figure 2: Two pairs of clean and noisy chunks illustrating the generation of training data for the paired-input DNN. In the positive pair, the distorted speech chunk was derived from the clean chunk through the non-linear distortion function $f(x(t))$; in the negative pair it was not.

rectified linear units (ReLU) [17] each, and a single sigmoid output unit predicting y_{ij}. It is initialized with random weights and trained using backpropagation with dropout of 20% of the hidden units [18] and ADA-Grad stochastic gradient descent [19]. Training minimizes the cross-entropy between y_{ij} and $g(z_j, x_i)$

$$\sum_{i,j} -y_{ij} \log g(z_j, x_i) - (1 - y_{ij}) \log(1 - g(z_j, x_i)). \quad (3)$$

3.2. Transition affinities

We also compute, $T(z_i, z_j)$, the affinity of transitioning from dictionary element z_i to dictionary element z_j. For now, this computation is not learned and only based on the acoustic features that are input to the neural network. We construct a matrix T_{ij} of all possible such transitions by comparing the log mel spectrum of the end of z_i to that of the beginning of z_j. In particular

$$T_{ij} = \exp\left(-d_\tau(z_i, z_j)/\gamma\right) \quad (4)$$

where $d_\tau(z_i, z_j)$ is the Euclidean distance between the log mel spectrum of the last τ frames of z_i and the first τ frames of z_j and γ is a parameter that controls the mapping of distances to affinities. Because our dictionaries contain approximately 60,000 elements, computing the whole transition matrix is feasible, but storing it is not. We thus store only the largest 10,000,000 (0.28% of) entries and set all others to be equal to the smallest stored entry.

In general, this transition model could take into account the compatibility of dictionary chunks according to many different types of comparisons. Such comparisons could include fine spectro-temporal details such as pitch, phonetic information like the identity of the current, preceding, and succeeding phonemes, and linguistic information such as an n-gram language model. While we did not include such characteristics in the transition model for the current experiment, it is easy to annotate the dictionary with them and compute compatibilities of dictionary chunks with these characteristics offline for use at test time.

4. EXPERIMENTS

The experiments were performed on the CHiME2-GRID small vocabulary dataset [20], which simulates read speech in a living room environment with recorded household noises. We do not use the

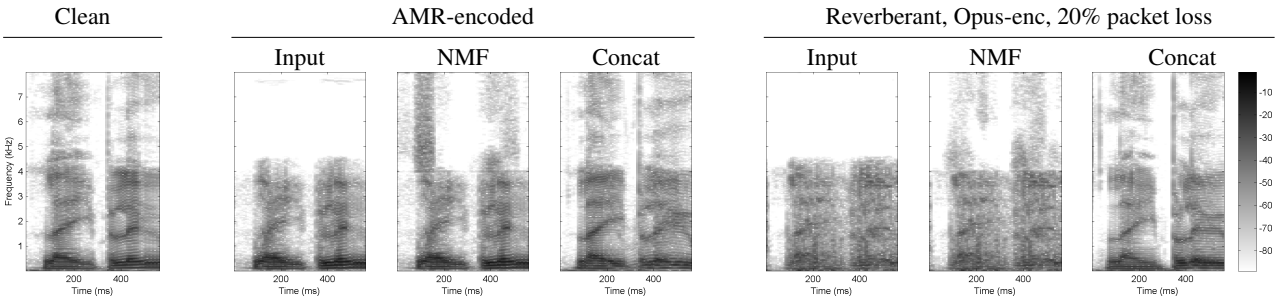


Figure 3: Example enhancements of the words "Lay blue" by the NMF and concatenative resynthesis (Concat) systems for two different types of inputs: AMR-encoded clean anechoic speech and Opus-encoded reverberant speech with 20% simulated packet loss.

noise for the current experiments, however. The dataset utilizes sentences from the GRID corpus [21], which consist of six short words of the form, "⟨*command*⟩ ⟨*color*⟩ ⟨*preposition*⟩ ⟨*letter*⟩ ⟨*digit*⟩ ⟨*adverb*⟩", for example, "place blue at A 9 again." Our experiments use the original anechoic clean speech as the reference signal and the clean spatialized speech from the "reverberant" condition with the mean of the left and right channels as the reverberant signal. We apply two low bitrate audio codecs to the signal, the Adaptive Multi-Rate codec (AMR) [22] used in GSM mobile phones and the Opus codec [23, 24], which incorporates Skype's SILK codec and Xiph.Org's CELT codec. We apply AMR coding and decoding using the reference implementation version 11.0.0 operating at a fixed bitrate of 4.75 kbps. We apply Opus coding, packet loss, and decoding using the reference implementation utility `opus_demo` version 1.1 at a fixed bitrate of 6.0 kbps. Note that allowing Opus to set the bitrate automatically results in an average bitrate of 12 kpbs, so forcing it to go significantly below that results in a noticeable degradation of quality.

To train the models and for the dictionary, we use 490 of the 500 training utterances from the official CHiME2-GRID train/test split, with the remaining 10 utterances used as a validation set. To test the models, we used 12 utterances from the official development set. There were no repeated sentences in the dataset, although different utterances of the same word appear in many sentences. Each sentence is approximately two seconds long, for a total of 16 minutes of training speech for each speaker.

From these utterances, which are sampled at 16 kHz, we compute the log mel spectrogram in Matlab using [25]. We found that chunks of 192 ms strike a good balance between generality and meaningfulness. For training, we extract chunks from the log mel spectrogram that overlap by 176 ms with their neighbors. This leads to a training dataset of 124,080 clean-noisy pairs of chunks, half matching and half not matching. The testing dictionary consists of 67,040 clean chunks. Our experiments are speaker-dependent, in that models are trained and tested on different utterances from the same speaker. We selected speaker 3, a male, because his speech had the longest duration. Processing a 2 second mixture using the proposed system took approximately 8 CPU-minutes on an Intel Core i5-2500 system.

4.1. Comparison system

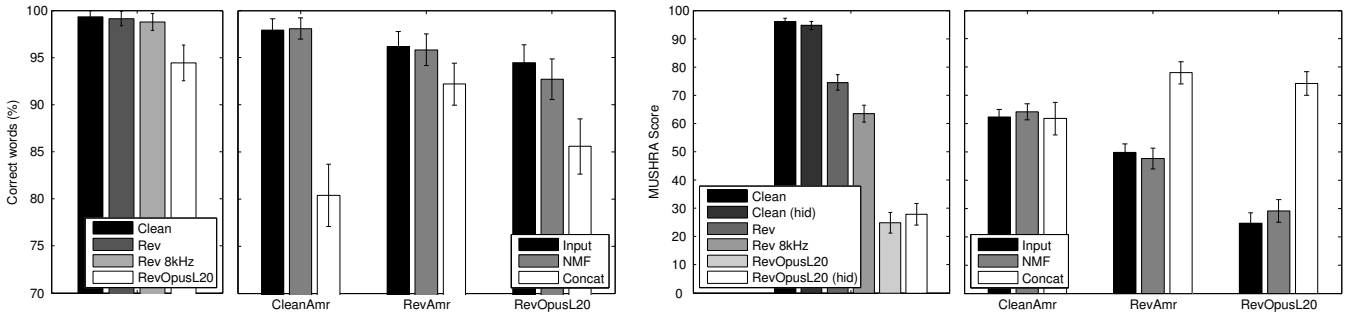
We compare our system against our own implementation of [11], another example-driven bandwidth enhancement system based on convolutive non-negative matrix factorization (NMF). This system factorizes the magnitude spectrum of input audio into separate source and filter models through liftering in the cepstral domain, and then models each of these signals using a separate convolutive NMF system. It learns a dictionary from wideband signals, identifies the activations using only the narrowband parts of the dictionary bases, and then reconstructs the wideband signal by combining these activations with the wideband bases. The phase of the high-frequency components is predicted separately from the phases of the narrowband components using linear regression. In our experiments, the dictionaries were learned from the same 490 utterances used to train our model, with the narrowband portion corresponding to the input speech and the upper frequencies to the output speech. We used the parameters described in [11], namely 50 envelope bases of length 8 frames and 100 excitation bases, each a single frame long. We use the convolutive NMF code of [26].

4.2. Listening tests

We conducted two listening tests to compare our system to the NMF system and various reference signals. One measured overall speech quality and the other measured intelligibility. In total, twelve versions of twelve different utterances were compared: the original clean speech sampled at 16 kHz (Clean), reverberant speech sampled at 16 kHz (Rev), reverberant speech resampled to 8 kHz (Rev 8kHz), and then three different input signals in their original form and processed by the NMF and concatenative resynthesis (Concat) systems. These inputs were the original anechoic speech encoded with AMR (CleanAmr), reverberant speech encoded with AMR (RevAmr), and reverberant speech encoded with Opus with a simulated 20% packet loss rate (RevOpusL20)[1]. See Figure 3 for excerpts from the beginning of several versions of one of the utterances.

The speech quality test compared these systems under a MUltiple Stimuli with Hidden Reference and Anchor (MUSHRA) paradigm. For each of the twelve original utterances, listeners were presented with the reference clean and distorted (RevOpusL20) utterance, and then with all twelve versions of that utterance unlabeled and in a random order. The utterances and comparisons were presented in a different random order for each listener. Listeners were asked to rate each utterance in terms of speech quality on a scale from 0 (poor) to 100 (excellent), where quality was defined in terms of suitability for being played over a living-room stereo system or being included in a radio broadcast. They were also instructed that an utterance could have high quality while having low intelligibil-

[1] These files can be heard at `http://mr-pc.org/work/waspaa15/`



(a) Intelligibility (b) Quality

Figure 4: Results of subjective listening tests showing (a) average percentage of words that listeners were able to correctly identify and (b) average MUSHRA score of each processed utterance (higher numbers are better). Errors bars show 95% confidence intervals. Left panels show various degraded signals, right panels compare enhancement systems on various input signals.

ity if it sounded like natural speech. Eight listeners participated in this test. None had heard the processed sentences before, although four had heard various subsets of the original GRID corpus and the CHiME2-GRID corpus.

The intelligibility test measured whether listeners' transcriptions of the different versions of each file matched that of the original sentence. Eight subjects evaluated these files, all of whom were native English speakers. They listened to all twelve versions of all twelve files in different random orders. Subjects were given a copy of the GRID grammar and asked to transcribe the sentences as best they could, noting that they did not necessarily have to adhere to the grammar. The GRID task itself only measures the accuracy on the letter and number in each sentence, the keywords, but here we report accuracy on all words. Accuracy on these keywords was comparable.

4.3. Results

Figure 4 shows results for all conditions averaged across utterances and listeners. These results show that the concatenative resynthesis system is able to significantly improve the speech quality of the degraded signals while the NMF enhancement of [11] is not. Intelligibility is already high for the speech degraded in various ways and there is no processing that increases its intelligibility, although the concatenative synthesis does significantly degrade intelligibility.

On anechoic speech encoded with the AMR codec (CleanAmr), the narrowband signals are relatively similar to the corresponding portions of the wideband signals, making them a good match for the NMF enhancement. These results are shown as the first group in the second panel of the intelligibility and quality plots, which show that the CleanAmr input signals have similar intelligibility and quality to the Rev 8kHz signal. Processing with NMF bandwidth expansion increases the quality of the signal slightly, but not significantly, and does not affect intelligibility. Processing with concatenative resynthesis also maintains the same quality, but reduces intelligibility to 80%. This is the worst condition for the concatenative resynthesis system, although it is not entirely clear why that would be the case. It appears that the decrease in quality and intelligibility was due to a large amount of double-talk in the resyntheses caused by incompatibilities between overlapping chunks.

The same system applied to reverberant speech encoded with AMR (RevAmr), which should be a more difficult condition, produced resyntheses with significantly higher intelligibility and quality. It also produced signals judged to be approximately 30 points higher quality than the original input and the NMF enhanced signal, which are statistically indistinguishable. The intelligibility of the concatenative resynthesis is 3-4 percentage points lower than the input signals and NMF, which is statistically significant, but not very large. On the reverberant speech encoded with Opus with a 20% packet loss (RevOpusL20), the concatenative synthesizer produces speech of the same quality while starting from significantly lower quality input. In this case, it produces an improvement of approximately 50 points over the original input, and approximately 45 points over NMF enhancement. Intelligibility for the concatenative resynthesis method is 85%, which is 10 percentage points below the input and 8 below the NMF enhancement.

On both of the reverberant inputs, because the concatenative resynthesis systems are mapping to anechoic clean speech, they can generate a much higher quality signal than the NMF system. Even if the NMF system's dictionary is built from clean speech, the activations that it will estimate for each basis will be fit to the reverberant signal and will thus reconstruct a reverberant wideband signal, assuming that the system is functioning perfectly. Furthermore, because the NMF enhancement system from [11] does not modify the narrow-band portion of the spectrum, it cannot restore damage to the spectrum from packet loss and coding like the concatenative resynthesis system can.

5. CONCLUSIONS

This paper has introduced a new approach to audio super-resolution that can not only increase the bandwidth of a speech signal, but also restore the quality of speech degraded by reverberation, speech coding, and simulated packet loss. These abilities derive from its use of a learned non-linear comparison function to drive unit selection in a concatenative synthesizer based on clean recordings. While the quality of the resulting resyntheses is quite high, it could be improved using more sophisticated approaches to synthesis, including the use of signal processing modifications to stitch adjacent chunks together and the use of a transition function able to utilize additional annotations of the dictionary elements. Such methods could also improve the intelligibility of the resulting syntheses, which is lower than that of the original speech mostly because of substitutions of one word for another and double-talk.

References

[1] S. Voran. "Listener ratings of speech passbands". In: *IEEE Workshop on Speech Coding for Telecommunications*. IEEE, 1997, pp. 81–82.

[2] E. Ekudden, R. Hagen, I. Johansson, and J. Svedberg. "The adaptive multi-rate speech coder". In: *IEEE Workshop on Speech Coding*. IEEE, 1999, pp. 117–119.

[3] N. Jayant and S. Christensen. "Effects of Packet Losses in Waveform Coded Speech and Improvements Due to an Odd-Even Sample-Interpolation Procedure". In: *IEEE Transactions on Communications* 29.2 (1981), pp. 101–109.

[4] A. Huggins. "Effect of lost packets on speech intelligibility". In: *NSC Note No. 78* (1976).

[5] M. I. Mandel, Y.-S. Cho, and Y. Wang. "Learning a concatenative resynthesis system for noise suppression". In: *IEEE GlobalSIP Conference*. 2014.

[6] Y. M. Cheng, D. O'Shaughnessy, and P. Mermelstein. "Statistical recovery of wideband speech from narrowband speech". In: *IEEE Transactions on Speech and Audio Processing* 2.4 (1994), pp. 544–548.

[7] P. Jax and P. Vary. "On artificial bandwidth extension of telephone speech". In: *Signal Processing* 83.8 (2003), pp. 1707–1719.

[8] H. Pulakka, U. Remes, K. Palomaki, M. Kurimo, and P. Alku. "Speech bandwidth extension using Gaussian mixture model-based estimation of the highband mel spectrum". In: *Proc. IEEE Intl. Conf. Acous. Speech & Sig. Proc.* IEEE, 2011, pp. 5100–5103.

[9] F. Nagel, S. Disch, and S. Wilde. "A continuous modulated single sideband bandwidth extension". In: *Proc. IEEE Intl. Conf. Acous. Speech & Sig. Proc.* IEEE, 2010, pp. 357–360.

[10] M. R. P. Thomas, J. Gudnason, P. A. Naylor, B. Geiser, and P. Vary. "Voice source estimation for artificial bandwidth extension of telephone speech". In: *Proc. IEEE Intl. Conf. Acous. Speech & Sig. Proc.* IEEE, 2010, pp. 4794–4797.

[11] D. Bansal, B. Raj, and P. Smaragdis. "Bandwidth expansion of narrowband speech using non-negative matrix factorization". In: *Eurospeech*. 2005.

[12] P. Smaragdis and B. Raj. "Example-driven bandwidth expansion". In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. 2007.

[13] P. Smaragdis, B. Raj, and M. Shashanka. "Missing data imputation for spectral audio signals". In: *IEEE Workshop on Machine Learning and Signal Processing*. IEEE, 2009, pp. 1–6.

[14] D. L. Sun and R. Mazumder. "Non-negative matrix completion for bandwidth extension: A convex optimization approach". In: *IEEE Workshop on Machine Learning and Signal Processing*. IEEE, 2013, pp. 1–6.

[15] B. P. Keegan, S. K. Tjoa, and K. J. R. Liu. "Super-resolution of musical signals using approximate matching pursuit". In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2011, pp. 81–84.

[16] S. Imai. "Cepstral analysis synthesis on the mel frequency scale". In: *Proc. IEEE Intl. Conf. Acous. Speech & Sig. Proc.* Vol. 8. IEEE, 1983, pp. 93–96.

[17] V. Nair and G. E. Hinton. "Rectified linear units improve restricted boltzmann machines". In: *Proc. ICML*. 2010, pp. 807–814.

[18] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. "Improving neural networks by preventing co-adaptation of feature detectors". In: *arXiv preprint arXiv:1207.0580* (2012).

[19] J. Duchi, E. Hazan, and Y. Singer. "Adaptive subgradient methods for online learning and stochastic optimization". In: *J. Mach. Learn. Res.* 12 (2011), pp. 2121–2159.

[20] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni. "The second 'CHiME' Speech Separation and Recognition Challenge: Datasets, tasks and baselines". In: *Proc. IEEE Intl. Conf. Acous. Speech & Sig. Proc.* 2013, pp. 126–130.

[21] M. Cooke, J. Barker, S. Cunningham, and X. Shao. "An audio-visual corpus for speech perception and automatic speech recognition". In: *The Journal of the Acoustical Society of America* 120.5 (2006), p. 2421.

[22] 3GPP. *Adaptive Multi-Rate Floating-point (AMR) Speech Codec*. Tech. rep. TS 26.104 V 11.0.0. 3GPP.

[23] J.-M. Valin, K. Vos, and T Terriberry. "Definition of the opus audio codec". In: *IETF, September* (2012).

[24] K. Vos, K. V. Sø rensen, S. r. S. Jensen, and J.-M. Valin. "Voice Coding with Opus". English. In: *Audio Engineering Society Convention*. Audio Engineering Society, 2013.

[25] D. P. W. Ellis. *PLP and RASTA (and MFCC, and inversion) in Matlab*. 2005.

[26] G. Grindlay. *nmflib: Efficient Matlab library implementing a number of common NMF variants*. 2010.