Deep Learning has become a popular approach for unsupervised feature learning [3]. It is now used extensively for object, face and speech recognition, as well as other classification tasks [1, 4]. The success achieved by deep learning is said to be due to the fact that these models comprise several layers of representation, with each higher layer expected to learn increasingly complex features of the given data. One example of such models is the Deep Belief Network (DBN) [5], a generative probabilistic model having a Restricted Boltzmann Machine (RBM) as fundamental building block. In particular, Convolutional Deep Belief Networks (cvDBN) have been very successful at audio classification tasks [2, 3]. In cvDBNs, the weights are shared between multiple units of each RBM [1, 2]. Convolutional approaches have been used for learning of invariant local features from high-dimensional data. Recently, cvDBNs have been shown capable of learning high-level features from audio spectrograms [3]. The cvDBNs map the input space to the space of latent variables by applying convolution of the audio spectrogram with the weight matrix. The convolutional networks in [3] use the entire spectrogram (following PCA whitening) as input to the RBM, thus requiring either the use of a fixed audio length or sampling that converts the time-domain signals into fixed-length spectrograms. Moreover, the length of the audio data is usually very large, which can make training [4, 5] computationally difficult in the case of big data applications. As pointed out in [3], it remains an interesting problem to apply deep learning to larger data sets and more challenging tasks. Further, we note that when applied to music genre classification, the first layer features performed the best overall [3] (using 300 nodes with a filter length of 10).

In order to be able to handle big data more efficiently, we propose the concept of convolutional data, that is, we investigate the use of a convolutional partition of the data as the input to a standard RBM. Convolutional data seeks to convert the high-dimensional spectrogram into chunks of lower-dimensional samples, while preserving their local spatiotemporal information, as done through weight-sharing in cvDBNs. Through data partitioning, audio data of different lengths can be used. As is [3], the audio is converted into spectrograms and PCA whitening is applied before the data is partitioned and fed into the RBM for training. Let \( X \) denote the input data spectrogram, \( N \) the window size, and \( t_{\text{step}} \) the step size within each spatiotemporal chunk. We make \( X_i = X(t:t+N) \) and \( t = t + t_{\text{step}} \) for each value of \( i \). Through the use of standard RBMs, rather than cvDBNs, and a quantization of the features learned by the RBM’s latent variables (similarly to the bag-of-words approach for language processing), we propose to build audio words which should give us an efficient algorithm for large-scale data. Finally, a classifier such as a Support Vector Machine (SVM) can be used on the audio words to produce, for example, genre classification in the case of music data (classical, blues, rock, etc.).

Figure 1: Convolutional Data for Audio Learning

As part of an initial experimental evaluation, we have used the GTZan genre dataset [6], with training, validation and test sets selected with a ratio of 2:1:1. An RBM containing 80 input nodes and 400 hidden nodes was trained on one-dimensional convolutional data created using a window size of 30 frames and a step size of 20. The audio words produced by the RBM were provided to an SVM for classification. For a two-genre classification task, an accuracy of 96% was obtained on the validation set, and an accuracy of 96% was also obtained on the test set. For a three-genre task, however, an accuracy of 74.6% was obtained on the validation set, and only 56% was obtained on the test set, a considerable drop in performance. This was to be expected given the loss of information resulting from the data convolution achieved as a tradeoff for efficiency. Experiments on the use of a stack of two RBMs for comparison are ongoing. In the case of a two-genre task, however, initial results are promising and the use of a single RBM seems to be sufficient. This opens up the possibility of efficient big data processing for personalized (yes/no) recommendation on resource-constrained mobile devices.

Although we do not expect to outperform the state-of-the-art in genre classification, we believe that this research can lead to new possibilities in the area of knowledge extraction [7] and understanding of audio data. As in [3], one can illustrate what the network has learned through visualization (see Appendix). In [3], first-layer bases are randomly selected for visualization. In [7], the bases with the highest information gain are used for knowledge extraction. To be continued.

1 We are grateful to Emmanouil Benetos for many useful discussions and feedback on this work.
Appendix – Network visualization
Visualization of 20 first-layer bases ($N = 30, t_{spp} = 20$). The frequency values are lower at the top. The top figure shows a visualization of a *blues* audio word; the bottom figure shows a visualization of a *classical* audio word.