A Simple Deconvolutional Mechanism for Point Clouds and Sparse Unordered Data (Student Abstract)

T. Paniagua,[1,2] J. Lagergren,[1,2] G. Foderaro[2]
{tapaniag, jhlagerg}@ncsu.edu, gfoderaro@ara.com
[1]North Carolina State University
Raleigh, NC 27695
[2]Applied Research Associates, Inc.
8537 Six Forks Rd., Suite 600
Raleigh, NC 27615

Abstract

This paper presents a novel deconvolution mechanism, called the Sparse Deconvolution, that generalizes the classical transpose convolution operation to sparse unstructured domains, enabling the fast and accurate generation and upsampling of point clouds and other irregular data. Specifically, the approach uses deconvolutional kernels, which each map an input feature vector and set of trainable scalar weights to the feature vectors of multiple child output elements. Unlike previous approaches, the Sparse Deconvolution does not require any voxelization or structured formulation of data, it is scalable to a large number of elements, and it is capable of utilizing local feature information. As a result, these capabilities allow for the practical generation of unstructured data in unsupervised settings. Preliminary experiments are performed here, where Sparse Deconvolution layers are used as a generator within an autoencoder trained on the 3D MNIST dataset.

1 Background

Generator mechanisms are an important component of many machine learning architectures, where they are used to transform compressed data into some expanded representation that is desired by the user. They are most commonly seen within approaches like autoencoders and generative adversarial networks (GANs) in the form of transpose convolutional or fully-connected feed-forward layers of increasing size. Much work has been done on these approaches, but the data processed (e.g. images) is always structured, as required by the models' components, restricting their applicability to standard domains.

Irregular and sparse data is found in many common system formulations, sensor outputs, and data sets. For example, point clouds are obtained from outputs of LIDAR and other 3D scans and have advantages over other 3D representations due to their compactness. However point clouds often have a variable number of points and lack structure, limiting their practical use by machine learning approaches. Adding edges or connections, meshes and graphs also share these qualities. Although it is believed that the approach presented here can be extended to many other data formats, potentially increasing the effective informational bandwidth of a large number of deep learning systems, this preliminary work is focused on point clouds.

While recent strides in the field of geometric deep learning have resulted in algorithms that can extract features from point clouds in supervised settings like segmentation and classification, little progress has been made toward the generation of such data in unsupervised settings like autoencoding. Solutions for point cloud upsampling, such as PU-Net (Yu et al. 2018), exist but do not support generation from a compressed latent feature vector representation of the point cloud. Therefore these methods do not support point cloud upsampling within generative models or for tasks such as unsupervised feature extraction. Some methods to generate point clouds from a latent space have been developed, such as (Zamorski et al. 2018), but they suffer from issues such as a high number of trainable parameters or the lack of local feature information used in the upsampling process. The Sparse Deconvolution presented here is a simple and scalable mechanism that is able to generate point clouds from compressed latent representations without ignoring local feature information.

2 Sparse Deconvolution Mechanism

The Sparse Deconvolution is a trainable mechanism that generalizes the transpose convolution operator to unstructured data. The operation uses a deconvolution kernel, which maps an input feature vector and set of trainable scalar weights to the feature vectors of m child outputs. The kernel consists of two multilayer perceptrons (MLPs), which both input a parent point's feature vector. The first MLP outputs the spatial offsets of the m child points relative to their parent point, and the second MLP outputs the feature vectors of each child point.

To concretely illustrate one Sparse Deconvolution operation, consider an arbitrary point $p(\mathbf{x}, \mathbf{z})$ in d spatial dimensions characterized by its position $\mathbf{x} \in \mathbb{R}^d$ and feature vector $\mathbf{z} \in \mathbb{R}^f$ that we wish to deconvolve into multiple child points. The Sparse Deconvolution mechanism is comprised of two MLPs, denoted by $h_{\mathbf{x}}(\mathbf{z})$ and $h_{\mathbf{z}}(\mathbf{z})$, which input the feature vector $\mathbf{z}$ and output the spatial offsets and feature vectors of the child points, respectively. The positional MLP

$h_{\mathbf{x}}$ returns m spatial residuals $\Delta\mathbf{x}_i \in \mathbb{R}^d$ relative to the parent, and the feature-generating MLP $h_{\mathbf{z}}$ outputs m feature vectors $\mathbf{z}'_i \in \mathbb{R}^f$, where i is the index of the i^{th} child point. Therefore m new child points, characterized by features $\mathbf{z}'_i$ and positions $\mathbf{x}'_i = \mathbf{x} + \Delta\mathbf{x}_i$, are generated from the parent point p. In this way, the Sparse Deconvolution is applied to all points of a latent point cloud to grow the set of points, and it is repeated iteratively until the desired number of points is achieved, just as transpose convolutions are applied to deconvolve images until a desired resolution is reached.

3 Preliminary Experimental Results

The Sparse Deconvolution mechanism is demonstrated here as the decoder stage of an autoencoder, which is a popular unsupervised approach for learning encodings of data. The objective of the model is to input a point cloud, compress it to a reduced latent representation, and reconstruct the original point cloud from that latent representation.

The autoencoder model was trained using the 3D MNIST dataset (Castro 2016), a 3D point cloud representation of the classic MNIST handwritten digit dataset. For the authors' convenience and faster training on limited hardware, the number of points in each sample was reduced to 2500 by randomly selecting from the original point clouds. The points are defined as coordinates in three-dimensional Euclidean space.

The autoencoder follows the standard architecture, but with each mechanism replaced by its point cloud analog, many of which are borrowed from other work. In the encoding stage, we extract a fixed size latent feature vector describing a point cloud of arbitrary size. This is done by applying the convolutional layer from (Simonovsky and Komodakis 2017) followed by the operation from (Dhillon, Guan, and Kulis 2007) that, similar to a pooling operation, eliminates redundant information after local feature extraction. These two operations are repeated iteratively, reducing the number of points by approximately half each time, where each child point cloud contains sufficient information to reconstruct their parent. After repeating these operations multiple times, global pooling is applied to generate a feature vector of fixed size, similarly to (Charles et al. 2017). Then the fixed-size latent feature vector is decoded by applying Sparse Deconvolutions iteratively until the size of the output point cloud matches that of the original. The reconstruction similarity is evaluated quantitatively using Chamfer distance, which is also used as the training loss.

The results are shown in Figure 1, where the original and reconstructed point clouds are displayed side-by-side. It is seen that the reconstruction assembles to form a representation of the appropriate digit as a smooth point cloud that is even more legible than the original.

4 Conclusion and Future Work

This work presents a Sparse Deconvolution mechanism that generalizes classical transpose convolutions to sparse unstructured domains. This method uses a simple deconvolutional kernel to achieve fast and accurate upsampling during generative tasks, which has been demonstrated with an autoencoder model trained on the 3D MNIST dataset. The authors believe that the Sparse Deconvolution mechanism is also extensible to other formulations, such as outputs of variable size without the need for expensive recurrent architectures and the processing of other data formats (e.g. meshes and graphs).

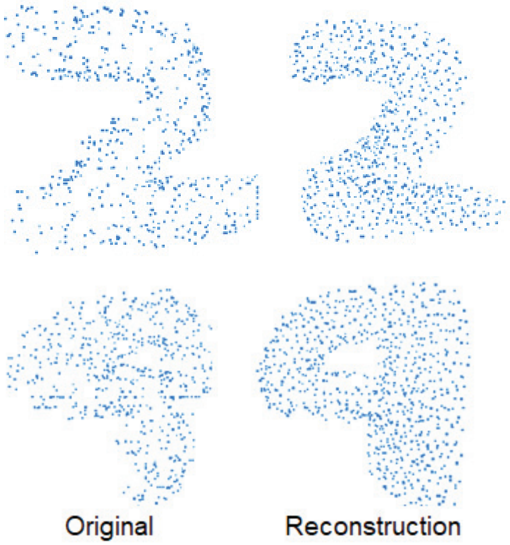

Figure 1: Example point cloud reconstructions using an autoencoder with Sparse Deconvolutions. Left: Original, Right: Reconstruction

5 Acknowledgements

This work was supported by the Defense Threat Reduction Agency and Applied Research Associates, Inc.

References

Castro, D. I. 2016. 3d mnist dataset. *https://www.kaggle.com/daavoo/3d-mnist*.

Charles, R. Q.; Su, H.; Kaichun, M.; and Guibas, L. J. 2017. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE Conf. on Comp. Vision and Pattern Recognition*, 77–85.

Dhillon, I. S.; Guan, Y.; and Kulis, B. 2007. Weighted graph cuts without eigenvectors a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(11):1944–1957.

Simonovsky, M., and Komodakis, N. 2017. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *IEEE Conf. on Comp. Vision and Pattern Recognition*, 29–38.

Yu, L.; Li, X.; Fu, C.; Cohen-Or, D.; and Heng, P. 2018. Pu-net: Point cloud upsampling network. In *IEEE Conf. on Comp. Vision and Pattern Recognition*, 2790–2799.

Zamorski, M.; Zieba, M.; Klukowski, P.; Nowak, R.; Kurach, K.; Stokowiec, W.; and Trzciński, T. 2018. Adversarial autoencoders for compact representations of 3d point clouds.