Proceedings of the NROCK 2023
The IV Nordic Symposium on
Rock Mechanics and Rock Engineering
ISBN 9xx-xxxx-xxx-x-z

Deep learning application in characterization and prediction of overbreak geometry in tunnels using point cloud data

Jorge Alvarez (jalvarez@srk.cl)

Jorge Alvarez
Sebastian Veloso,
Esteban Hormazabal
SRK Consulting Chile, Santiago, Chile

ABSTRACT

An overbreak during the construction of underground mining tunnels is a common geotechnical and operational problem, which is caused by a combination of geological, geotechnical, structural and operational factors, in which partial or reduced information is available, thus conditioning tunnel stability and consequently the safety of personnel during construction. Additionally, studying an overbreak during early stages allows to validate assumptions applied during engineering stages.
Throughout history, different methods have been proposed for the overbreak estimation, these ranging from an empirical, analytical (including numerical modelling), observational or even through the application of machine learning.
This work proposes a different approach to most of the studies carried out, which usually consider an average or expected value of overbreak. On this occasion, Deep Learning architectures are used to characterize and predict the complete geometry of the tunnel based off of a training carried out using point clouds of the sectors already excavated.
The results obtained show that it is possible to use autoencoder-type architectures to carry out the characterization and prediction of the tunnel's geometries from point clouds of previously excavated sectors, which has a relevant value for back analysis and potentially predictive analysis, which would in turn impact tunnel stability and/or safety in the different operation cycles during the construction of underground mining galleries and/or tunnels and civil works' projects and operations.

KEYWORDS

Tunnels; Overbreak; Point Cloud Data; Deep Learning; Autoencoders

INTRODUCTION

Overbreak is the zone around the tunnel with a larger cross-section than necessary according to the original design. This has important implications in terms of stability, costs and construction time in civil and mining projects. Although different methods have been used to address this issue, a definitive solution has not been found due to the difficulty of establishing a clear relationship between overbreak and its causes. In this paper, an approach is presented in which the complete 3D geometry of the excavated tunnel is used and artificial intelligence techniques, such as Deep Autoencoder Network (Vincent P., et al., 2010) are applied to characterize overbreak geometries and potential prognosis.

The central idea associated with the characterization of overbreak geometries is that during the engineering stages of a civil or mining project, different assumptions are established as input parameters, which are later evaluated during the construction process, which can result in significant differences between the expected behavior according to design and reality. Given the above, the application of an artificial intelligence algorithm oriented to the geometric segmentation of a tunnel aids for progress in terms of a zoning tool, which added to the design assumptions and as-built conditions, enables new forms of analysis. Figure 1a), presents the proposed analysis outline. For example, in Figure 1b), the geometric shape of a tunnel is conditioned by the insitu stress field, activating zones of potential spalling towards one side or the other according to the orientation of the principal stresses (Wahid Ali, et al., 2022). This information allows validating or refuting the expected behavior according to design, thus when a contradiction is observed, it is possible to recommend in this particular case, further stress measurements in the area to corroborate the phenomenon or the application of mitigation measures such as a different support design, but in general terms it translates into the need for a better characterization of the potential causes (validation of assumptions).

A second point to assess would be the possibility of predicting the overbreak associated geometry (see Figure 2a). The central concept would be applying an artificial intelligence algorithm in the horizontal progress cycle of a tunnel using, for example, the drill and blast method. As shown in the figure aforementioned, depending on the prediction, an assessment of the potential risks could be carried out and subsequently the necessary mitigation measures taken; modifying the arrangement of the advance per blast or other support alternatives. Additionally, this would also enable the possibility of estimating those areas that cannot be surveyed by 3D scanner, as shown in Figure 2b, since it is not uncommon that some areas cannot be surveyed topographically for operational reasons, for example, areas with debris that would prevent the completion of the surveys.

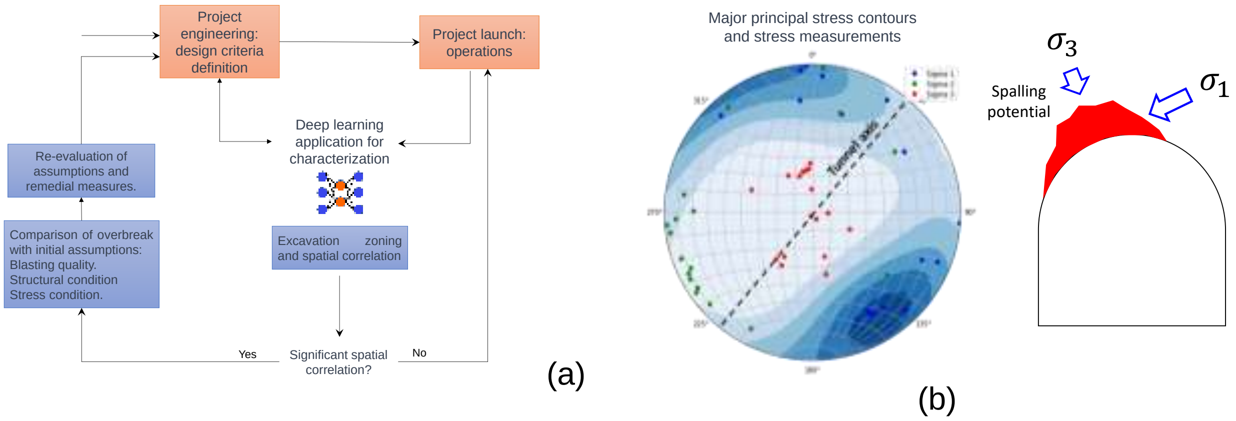


Figure 1. Application scheme of a characterization algorithm. (a) Explanatory diagram of a characterization algorithm applied in a UG project. (b) Example of a geometric characterization cross-referenced with stress measurements application.

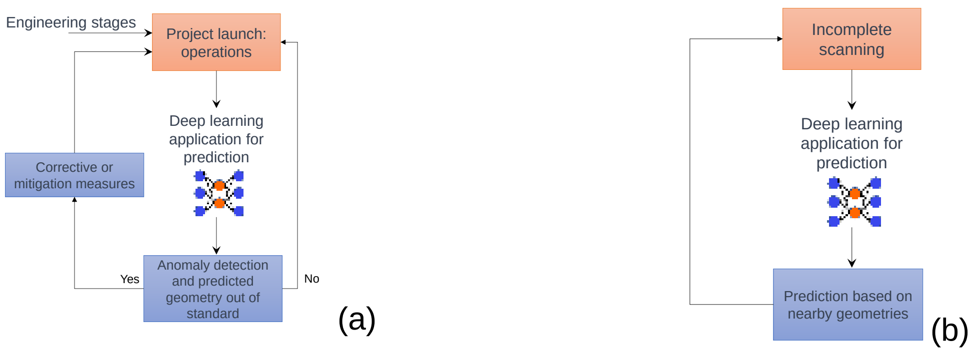


Figure 2. Application scheme of a prediction algorithm. (a) Explanatory diagram of a prediction algorithm applied during the horizontal advance per blast cycle (b) Explanatory diagram of an incomplete scanning estimation.

1. THEORETICAL DEFINITIONS

The following are some basic definitions for the development of a deep learning algorithm to characterize and predict overbreak using a complete geometry.

1.1. Machine Learning, Deep Learning and Autoencoders

Machine Learning and Deep Learning correspond to subfields of Artificial Intelligence (AI), while Machine Learning is the study of algorithms that improve task performance through experience making decisions without being explicitly programmed, Deep Learning, a subarea of Machine Learning, is often better suited for Big Data problems and has generated a paradigm shift in attribute extraction and compositional layer learning (Zhang A., et al., 2022).

Pattern discovery is an important task that allows to obtain useful information from large datasets (Alla & Kalyan Adari, 2019). One way to do this is by using autoencoders, which, in general terms, is a type of unsupervised neural network that aims to compress input data into a representation called latent space, which has a much lower dimensionality than input and that is characterized by preserving the most relevant information in an intelligent way. The main objective is to reconstruct the input from the latent space, seeking to minimize an error function. This can be thought of as a nonlinear version of principal component analysis or matrix factorization. In detail, autoencoders are divided into two components: one that encodes the data in the latent space and a decoder that reconstructs it from this encoding. Autoencoders can be used to reduce the complexity of the data and learn more abstract representations of it, which can be useful in various machine learning tasks. A basic architecture of an autoencoder can be seen in Figure 3.

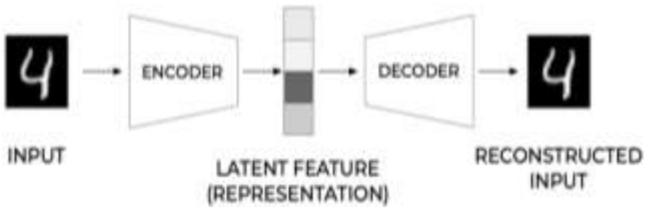


Figure 3. Typical autoencoder diagram (taken from Michelucci U., 2022).

1.2. Point Cloud Data Models

Artificial intelligence techniques for dealing with point clouds have evolved over time from the use of voxel grids for a structural treatment to architectures that are point cloud order invariant such as PointNet and PointNet++ (Charles R. Qi., et al, 2017). PointNet can handle cluttered point clouds and can be trained to classify, segment, and semantically analyze sets. PointNet++ uses a hierarchical architecture to improve local pattern capture (see Figure 4).

Other architectures such as FoldingNet (Yaoqing Yang, et al., 2018) also exist, which corresponds to a form of autoencoder, capable of transforming a 2D to 3D dimensional mesh. The input to the encoder is an n by 3 matrix, where each row of the matrix consists of the spatial position in three dimensions (x, y, and z). The output is an m by 3 matrix, representing the reconstructed point cloud.
Chamfer distance can be used to measure the reconstruction error, which measures the distance between the original and the reconstructed point cloud. The autoencoder computes a latent representation of each input point cloud and subsequently reconstructs the point cloud using this representation. Other architectures for point cloud processing, which have not been used in this work, include PCN and PointConv++ (Qi Yang, et al., 2021) which have used the PointNet and PointNet++ algorithms as their basis, respectively.

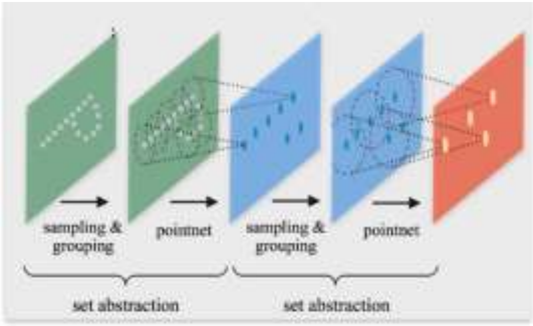


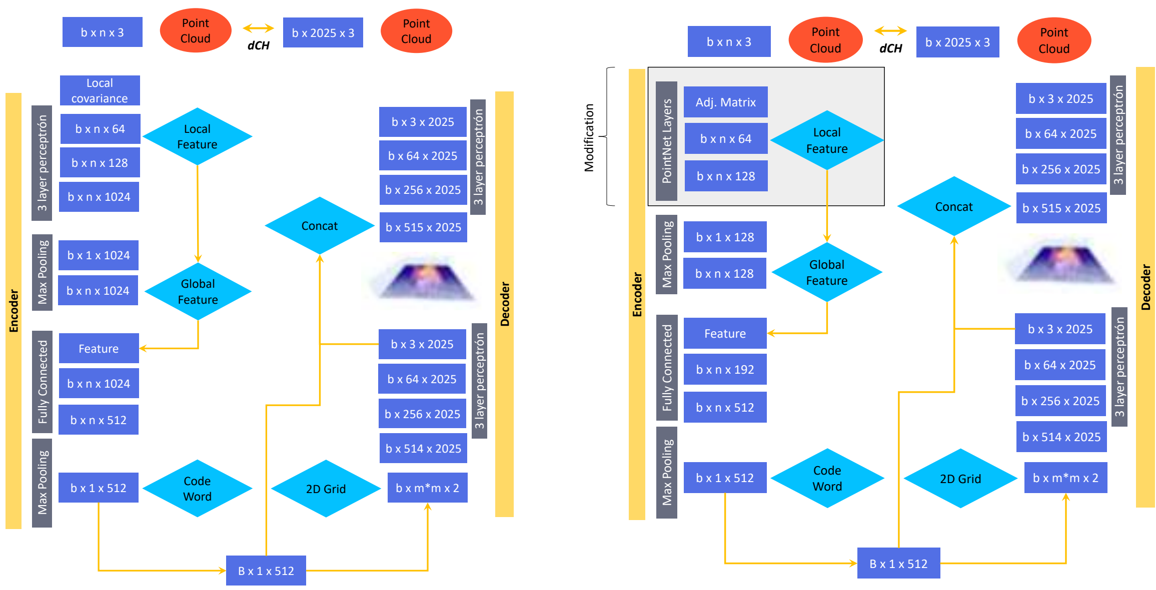
Figure 4. PointNet++ diagram, for capturing local and global attributes in a point cloud (taken from Charles R. Qi, et al., 2017).

2. MODEL SELECTION

FoldingNet base architecture is considered in terms of characterization and a modified architecture as an additional alternative, where a hierarchical extraction of both local and global attributes is incorporated through the use of PointNet++.

To evaluate the efficiency in the recognition of representations in the latent or compressed space (called code word), a public dataset is used where the different geometric categories to be recognized are known in advance like the ShapeNet dataset (Chang, et al., 2015) contains close to 17,000 3D point clouds with 16 different categories. The procedure consists of using the codeword of the latent space and then grouping in "n" clusters, to then compare these groupings with the actual categories of the dataset. Figure 5 shows the two architectures evaluated.

The results obtained indicate that the representativeness of the geometries in the latent space is higher with the modified architecture compared to the base, especially when the number of categories is lower. Therefore, a modified architecture is selected for the characterization task.



(a) (b)

Figure 5. Deep learning neural architectures. (a) FoldingNet-type base architecture, (b) Modified architecture including PointNet++ (modified from Yaoqing Yang, et al., 2018).

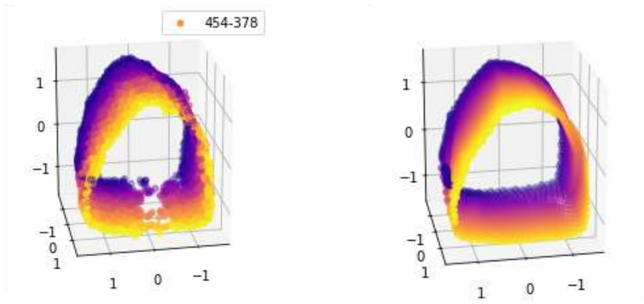
Table 1. Geometry recognition comparison performance in latent space.

N° Categories	Base Architecture Accuracy (%)	Modified Architecture Accuracy (%)
4	84	99
6	81	96
8	79	82

3. CHARACTERIZATION RESULTS

3.1. Typical Outputs

Figure 6 displays a typical output using a modified architecture. The reconstructions obtained show a "clean" point cloud without disturbances or noise, and also allow extrapolating areas where the original point cloud has missing data.



(a) (b)

Figure 6. Typical output from the model. (a) Input geometry in a model, (b) Output geometry of the model.

3.2. Results

After inserting 400 geometries corresponding to tunnel segments to the modified architecture and applying the codeword or latent space representation, it is possible to visualize each of these geometries in a 2 dimensional space by means of the commonly used PCA (principal component analysis). The graph on Figure 7, in addition to facilitating the visualization of the total geometries, has a logical order associated with the learned characteristics of the data set. The cluster with the highest density was sought using a DBSCAN type algorithm (Pedragosa F., et al., 2011) which indicates which geometries are repeated in greater quantity and have a certain degree of similarity. The exploration of the latent representations indicates that most of the geometries are within standard according to the design geometry (see Figure 7-1), while those far from the zone of higher density have biases in their shape (see Figure 7-2, 7-3 and 7-4), with the overbreak controlled by the presence of structural planes or joints in the roof area.

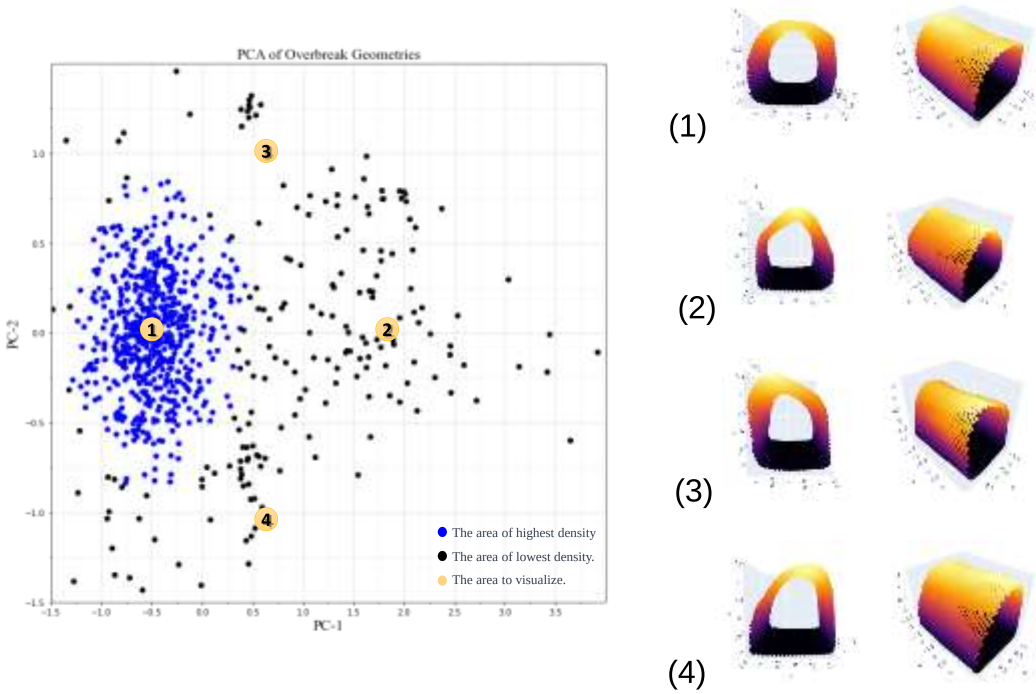


Figure 7. Recognition of overbreak features in latent space and their exploration in different zones.

Finally, it can be indicated that, from the observation of the latent space through a reduction of dimensionality, it is possible to characterize the overbreak associated geometries. Additionally, it can be seen that there is a logical order in the arrangement of the geometries, which can later be zoned in a layout of an underground project and/or allow a back- analysis of the initial design assumptions.

4. PREDICTION RESULTS

4.1. Geometry processing

For the overbreak prediction objectives, the same previous architecture was considered, with the exception that the data loading considers providing a point cloud used as input, which corresponds to a previous section of the tunnel to be predicted.
Figure 8, displays the difference between the data load associated with characterization and prediction.

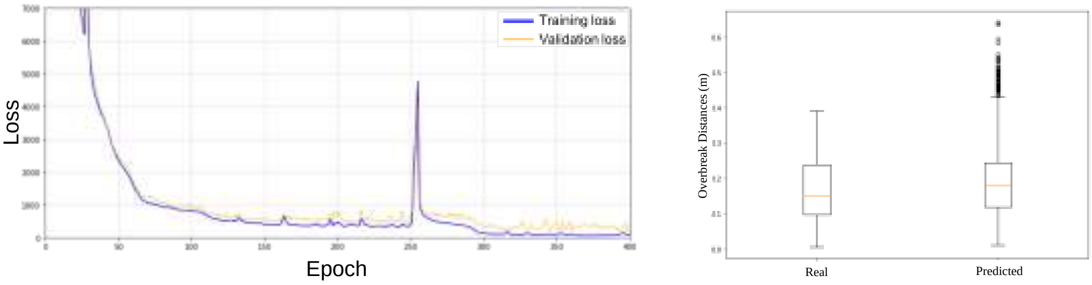
(a) (b)

Figure 8. Data loader difference for prediction task and characterization task. (a) Data loader for characterization, (b) Data loader for prediction.

In this case, since the objective is not the characterization of geometries, the data processing does not consider a standardization of the points. A transformation of coordinates to the origin and a rotation were considered so that all geometries are aligned in the same direction, taking advantage of the largest amount of available data from excavations with different orientations.

4.2. Results

The results obtained indicate that there is no evidence of over-fitting in the training process, since both the training and validation loss curves decrease in a similar way and without separating from each other as the training is carried out (see Figure 9a). On the other hand, when reviewing a prediction of the validation set, a good agreement is seen in terms of distribution of the over-excavated distances, with an actual median of 0.16m and a predicted median of 0.19m, but with a higher variability (outliers). The association between the prediction and the actual point cloud can be seen in Figure 9b.



(a) (b)

Figure 9. Training metrics. (a) Loss curve of the training process (b) Overall comparison of validation, prediction of overbreak and actual overbreak.

The metrics associated with the loss function in the training process are the chamfer distance (CD), which measures the dissimilarity between two point clouds by summing the distances between corresponding points in each cloud, and the mean absolute error (MAE), and are presented in Table 2:

Table 2. Metrics associated with the prediction target.

Set	**CD Loss** (m^2)	**EAM** (m)
Training	57	0.18
Validation	118	0.21

Although the indicators seem to be favorable, there is still margin for result improvement, as can be seen in a validation example (see *Figure* 10), although there is a general agreement between the predicted and the real point cloud, in certain specific sectors the prediction level responds better than in other sectors. For example, in the roof area, at point A, there is a geometric depression that is captured by the model during the prediction, however, when moving to the right with point B, the major overbreak in the roof area is not captured.

As it is possible to observe, using a point cloud as input permits the learning of the neural network, furthermore, the prediction made adjusts quite well to the actual point cloud in terms of average overbreak, considering that causal factors/attributes such as: rock quality, structural condition, blasting quality among others, which all correspond to variables commonly used for a prediction task, were not used therefore these could enrich the results of an AI model.

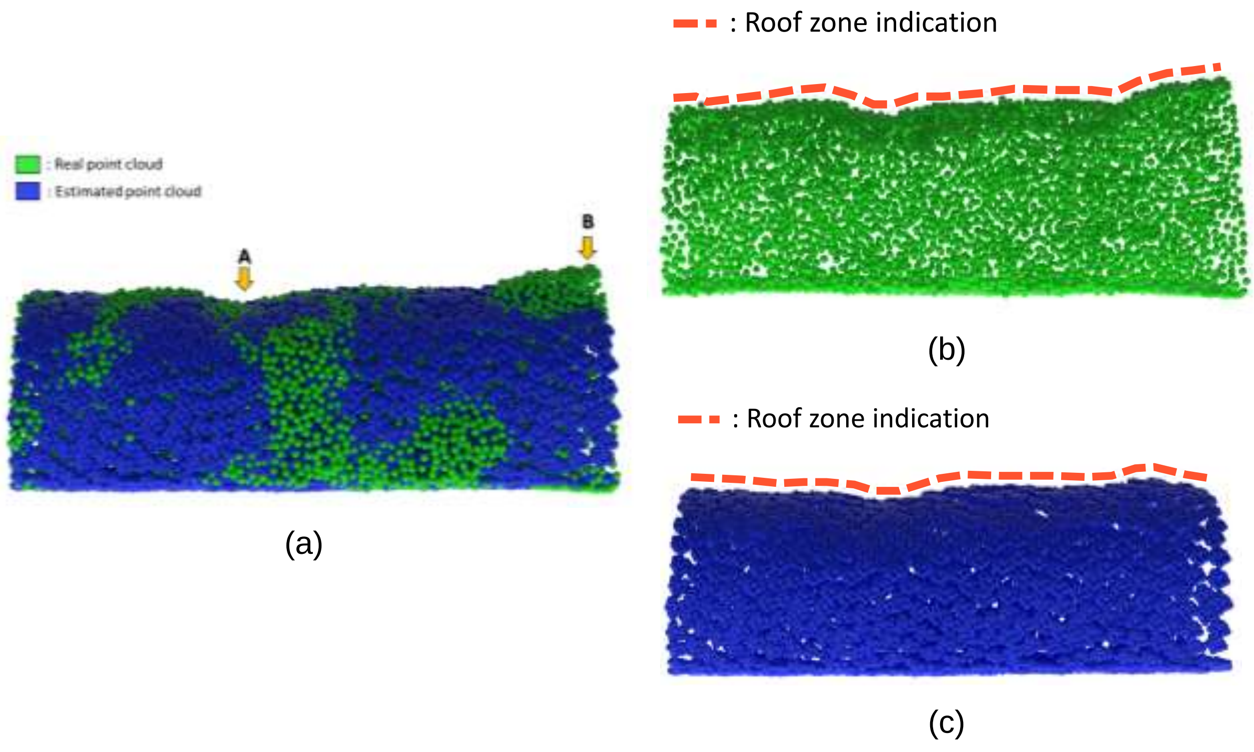


Figure 10. Tunnel Profile view. (a) Real and estimated geometry point cloud (b) Real geometry point cloud with roof zone indication (c) Estimated geometry point cloud with roof zone indication.

5. DISCUSSION OF RESULTS

Regarding the evaluation of architectures for characterization:

- Both base (FoldingNet) and modified (FoldingNet plus PointNet++) architectures were evaluated using the ShapeNet public dataset. The PointNet++ modified architecture performed better however, further performance evaluations with more batches and different public datasets are recommended to avoid a biased decision.
- The application to characterize overbreak geometries is effective and is simple to use, with interpretable results allowing to visualize the reconstructed geometries from the latent space.

Regarding the prediction of overbreak:

- The results show that a tunnel point cloud can be predicted when using another as input, and FoldingNet is flexible for this task. Overfitting was not observed for the prediction model and the prediction model could most likely be further improved after including causal factors/attributes such as structural condition, stress condition or blasting quality, among others.
- For this exercise geology and structural domains were not available so to improve the estimation, key factors/parameters such as structural conditions, in situ stresses, lithology or blasting quality, among others should be taken into consideration for future studies.
- Although the prediction model gave good results, it is recommended to evaluate alternative architectures that better capture local and global attributes of the input geometry, such as PointConv++ architecture. Additionally, the use of a PointMask-type architecture is suggested (Saeid Asgari Taghanaki, et al., 2020) to find those critical points that affect the output geometry, this in order to improve the interpretability of the results.

Applicability:
In terms of applicability AI models associated with characterization and prediction tasks, potentially applicable scenarios are:

Table 3. Potential applicability of AI algorithms in characterization and prediction.

Applications	When is it applicable?
Characterization	• For the purpose of a back-analysis of design parameters. • Apply when there is abundant scanning of excavations that have not been processed for analysis.
Prediction	• During the horizontal development cycle of a underground excavation. • When an estimate of zones with incomplete scanning is required.

6. CONCLUSIONS AND FUTURE WORK

Several Deep Learning algorithms for 3D point cloud analysis are currently available. This approach has great potential in areas of engineering, where geometries can enhance value. The modified FoldingNet model performed best for the task of characterizing overbreak geometries, based on the ShapeNet dataset and using a 4-element batch size. Autoencoders, especially FoldingNet, can be used to reproduce clean tunnel geometries that have incomplete zones.

Overbreak geometries point clouds can be predicted with other point clouds as input, which could be improved after including causal factors/attributes such as rock quality, blast quality, stress and structural conditions, among others.

In the field of geomechanics and particularly in overbreak analysis, AI can help us to discover patterns and features in the data more efficiently and with acceptable levels of accuracy, which can be considered as a tool for decision making.

Further performance evaluation of architectures using different public datasets, e.g., with ModelNet10 or ModelNet40 and with different batch size configurations, is recommended in order to detect possible biases in the model selection.

Regarding the characterization, an anomalous behavior detection analysis should be performed, in order to obtain a model that allows to automatically recognize those geometries that deviate from the ideal design. The incorporation of a variational autoencoder, which allows to generate a better distribution of geometries in the latent space and with the possibility of generating synthetic geometries for predictive numerical stability analysis (2D and 3D) could be considered.

In order to obtain a better interpretation of the inner workings of the neural network, an analysis of critical points should be incorporated to determine those geometric zones of the sections of the underground excavation that condition the results of the model. Regarding predictions, a larger data set should be used for the training process, as well as the incorporation of other explanatory variables and/or the evaluation of different architectures that better capture the local and global attributes of a point cloud.

7. ACKNOWLEDGEMENTS

The author would like to thank the various professors at Universidad Católica de Chile (Hans Lobel and others) and professionals at SRK Consulting Chile for their invaluable collaboration in the development of this research project and their generosity in providing information and giving their time for discussion and review of this work.

8. BIBLIOGRAPHY

Ali, W., Rehman, H., Abdullah, R., Xie, Q., & Ban, Y. (2022). Topography induced stress and its influence on tunnel excavation in hard rocks - a numerical a approach. *GEOMATE*, 9.

Alla, S., & Kalyan Adari, S. (2019). Beginning anomaly detection using Python-Based Deep Learning. Apress.

Asgari Taghanaki, S., Hassani, K., Kumar Jayaraman, P., Hosein Khasahmadi, A., & Custis, T. (2020). PointMask: Towards Interpretable and Bias-Resilient Point Cloud Processing. *ICML 2020*.

Chang, Angel X. and Funkhouser, Thomas and Guibas, Leonidas, & et al. (2015). ShapeNet: An Information-Rich 3D Model Repository. Chicago.

Charles R. Qi, Hao Su, Kaichun Mo, & Leonidas J. (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *CVPR 2017*.

Charles, R. Q., Yi, L., Su, H., & J. Guibas, L. (2017). PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. *Computer Vision and Pattern Recognition*.

Fey, M., & E. Lenssen, J. (2019). Fast graph representation learning with Pytorch Geometric. *ICLR*, 9.

Michelucci, U. (2022). *An Introduction to Autoencoders.* Retrieved from Toelt.ai.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel , V., Thirion, B., Grisel, O., & et al. (2011). Scikit-learn: Machine Learning in Python. *Machine Learning Research*, 6.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Antoine Manzagol, P. (2010). Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of Machine Learning Research 11*, 38.

Yang, Q., Lin, Y., Li, J., Wang, Y., & Yang, F. (2021). PointConv++: A muti-resolutional network for point cloud classification. *Association for the Advancement of Artificial*.

Yang, Y., Feng, C., Shen, Y., & Tian, D. (2018). FoldingNet: Point Cloud Auto-encoder via Deep Grid Deformation. *CVPR*.

Zhang, A., C. Lipton, Z., Li, M., & J. Smola, A. (2022). *Dive into Deep Learning.*