An Investigation of Semantic Segmentation on AI4MARS Dataset: A Project in Human Computation and Citizen Science

Zhe Fan, zhe.fan@mail.mcgill.ca
COMP 596 Project Report
Student ID: 260836923

Introduction

Human Computation is a branch of computing that involves recruiting human intelligence to complete complex and resource-intensive computational tasks. The collection of landscape segmentation data on the terrains of Mars, for example, is one such task. The AI4Mars dataset consists of around 326000 labels for 35000 images collected by the Mars Exploration Rover mission (MER) (Swan et al., 2021). The AI4Mars Dataset is annotated via crowdsourcing, and each image was labeled by ~ 10 people to ensure agreement and quality control (Swan et al., 2021). In this study, a U-Net, as well as an FCN-Resnet model, are trained on a subset of the AI4Mars dataset to evaluate the performance of deep-learning models on the crowed sourced semantic segmentation training data (Long et al., 2015; Ronneberger et al., 2015). The performance of the models will be analyzed by taking their 3-fold cross-validation f1-scores, as well as their f1 score metric on a gold-standard test set.

Methodology

Training Dataset Preparation. The dataset used to prepare for semantic segmentation training is the AI4Mars dataset released publicly on Kaggle by Swan et al. (2021). The default training + validation set size will be 2000. This set consists of the images and crowdsourced labels from the *edr* directory. Each image in the dataset contains up to 5 different terrain types – *soil, bedrock, sand, big rock,* and *others*, which serve as labels for the image pixels. An alternative training + validation dataset consisting of only 1000 examples is compiled for the analysis of the effect of data size on model performance.

Testing Dataset Preparation. Swan et al. (2021) provided three sets of gold-standard labels for a subset of 322 images in the AI4Mars dataset. Each set differs in the minimum number of agreements required to determine each pixel's label. (e.g. 3-label implies minimum 3 expert agreements per pixel label)

U-Net Model Preparation. The U-Net model by Ronneberger et al. (2015) is a popular model for biomedical image segmentation. The U-Net model's architecture consists of a contracting path (encoding) and an expansive path (decoding): during the contracting path, the input image is encoded through repeated application of two consecutive 3*3 convolutional kernels followed by shrinking with 2*2 max pooling steps (Ronneberger et al., 2015). During the expansive path, the input tensors are expanded through 2*2 up-convolutions followed by two consecutive 3*3 convolutional kernels (Ronneberger et al., 2015). For the AI4Mars dataset, the U-Net model is set up to output 5 output channels, each corresponding to a terrain category. The implementation of the U-Net model is adapted from the *Pytorch-UNet* repository by Milesi Alexandre (2023).

FCN Model Preparation. The Fully-Convolutional Network (FCN) model by Long et al. (2015) is another powerful model for semantic segmentation. The FCN model is "fully convolutional," meaning that it takes an input image of any arbitrary dimensions and contracts it into a coarse output tensor via a sequence of convolutional and pooling steps before the

deconvolution step (Long et al., 2015). The model also contains residual blocks which modifies its later network layers with its earlier layers' states (Long et al., 2015). The FCN implementation on PyTorch is inspired by the implementation provided by the PyTorch Developers (PyTorch, 2017).

Training and Validation. The models are trained and validated using 3-fold cross-validation on the training set using 2 * P100 GPUs. Thus, for a set of 2000 training images, the validation set would contain 667 images, and the training set would contain 1333 images. The splitting is repeated three times, such that the validation set for each fold would contain the first third, middle, and the final third of the training images. The number of epochs is set to 10, the optimization method is Stochastic Gradient Descent with a learning rate of 0.05. The performance of the model during each fold is evaluated with the validation set via the weighted f1 score metric:

$$f1\_class = TP\_class/(TP\_class + ½(FP\_class + FN\_class))$$

$$f1\_weighted = 1/(number\ of\ classes) * \sum_{each\ class\ i} f1_i * support_i$$

where *class* represents the class for which a particular metric is computed, *TP* represents the number of true positives, *FP* represents the number of false positives, *FN* represents the number of false negatives, and $support_i$ represents the proportion of occurrences of class *i* compared to all classes. Furthermore, the confusion matrices of each model's performance on the validation/test sets are computed.

Testing. The best-performing model (based on weighted f1 score) of each 3-Fold cross-validation fold will be compared against the gold standards. Weighted f1 scores and confusion matrices will be obtained for each model.

Results

3-Fold validation performance of the U-Net and FCN models (default training size). The weighted f1 scores of the U-Net model in their corresponding folds are 0.624, 0.581, and 0.653. The average weighted f1 score of the model is 0.619. The weighted f1 scores of the FCN model in their corresponding folds are 0.824, 0.830, and 0.830. The average weighted f1 score of the model is 0.828. The confusion matrices of the corresponding validation folds are provided in the appendix.

Performance on the gold-standard-test sets. The best models during the cross-validation experiments are selected to perform test validation on each of the three-gold-standard test sets (Table 2). Sample image predictions using the best-performing model are shown in Figure 1.

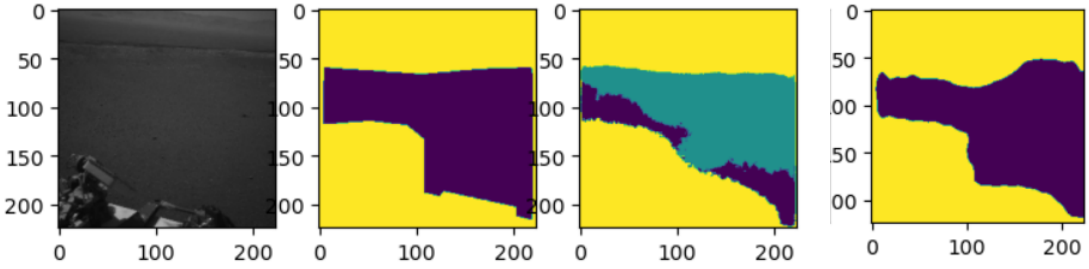

Figure 1. From left to right: 1. Original MSL Image; 2. Gold-Standard Labels (minimum 1-agreement); 3. UNet Predictions (default training size); 4. FCN Predictions (default training size).

Average K-Fold Validation Performance based on training set size. The weighted f1 scores of each model training using a different number of training samples (Table 1).

	Training size = 667 (Small)	Training size = 1333 (**Default**)
UNet	0.542	0.619
FCN	0.808	0.828

Table 1. The average weighted f1 scores of U-Net and FCN models on the validation folds.

Gold standards performance based on training set size. The weighted f1 scores of each model on the three gold stand test sets.

UNet	Small Training Set	Default Training Set	FCN	Small Training Set	Default Training Set
1-label	0.607	0.718	1-label	0.707	0.730
2-label	0.560	0.691	2-label	0.778	0.792
3-label	0.519	0.645	3-label	0.803	0.806

Table 2. Weighted F1 Scores of U-Net and FCN models on gold-standard test sets (organized by training set size). Each pixel of the test set images is labeled and determined by a minimum number of expert label agreements, ranging from 1 to 3.

Actual/Predicted	Soil	Bedrock	Sand	Big Rock
Soil	2571776 **(81.9%)**	133997	433034	0
Bedrock	141751	1147865 **(88.0%)**	14927	0
Sand	19498	80550	742235 **(88.1%)**	0
BigRock	241	1041	119	0 **(0%)**

Table 3. U-Net confusion matrix (with 3-label test set), the accuracy of the class prediction is bolded along the diagonal.

Actual/Predicted	Soil	Bedrock	Sand	Big Rock
Soil	3389974 **(92.3%)**	162837	118330	129

Bedrock	65063	2377030 **(96.8%)**	9908	3874
Sand	6189	190692	1556271 **(88.7%)**	1546
BigRock	1494	4750	266	1340 **(17.1%)**

Table 4. FCN confusion matrix (with 3-label test set), the accuracy of the class prediction is bolded along the diagonal.

Discussion

In this study, the training data set is partitioned into a training set and a validation set through a 3-fold cross-validation procedure. Although K-fold cross-validation is a robust and reliable method of measuring model performance through repeated sampling of the training dataset, it can also be very time-consuming and memory-demanding to perform as multiple models need to be prepared for different fold partitions. Alternative issues may arise due to an imbalance of label distributions between the folds: for example, a training fold with data distribution that has low variance may lead to an underfitting model. Thus, the model's average performance may be skewed by inordinately strong or poor validation performances on particular folds.

The U-Net model (default training size) achieved a weighted f1 score of 0.645 on the 3-label test set, while the best-performing FCN model achieved a weighted f1 score of 0.806 on the 3-label test set (Table 2). Note that a weighted f1 score ranges from 0 to 1, with 1 indicating a perfectly correct model. Swan et al. (2021) mention that the 3-label test set should be regarded as the most "confident" test set as it is generated through at least three experts' agreements. Thus, it is clear that the FCN model outperforms the U-Net model under the same hyperparameter settings on the 3-label test set. There may be several reasons to account for this: first, the FCN model contained around 2 million (6%) more trainable parameters than U-Net, which may improve its ability to capture underlying patterns in the training set. Furthermore, the FCN model contained a ResNet backbone, which allows for residual learning (He et al., 2016). Because both U-Net and FCN encode input image data through a "contracting process," the feed-forward residual structure of the FCN model may compensate for potential information loss due to a contracting network, leading to more accurate predictions and a better f1 score.

Swan et al. (2021) provide confusion matrices of predictions made by a DeepLabV3+ model (with a ResNet-101 backend pre-trained on ImageNet) on the 3-label test set. The group achieved a 96.67% overall accuracy on class label predictions; class-wise, the accuracy was 99.1%, 94.9%, 93.45%, and 93.24% for *Soil, Bedrock, Sand,* and *Big Rock*, respectively. Our best-performing model is the FCN model, which achieved an average accuracy of 73.7% (Table 4). However, our models performed poorly on the *Big Rock* class, while the average accuracy of the three other classes is 92.6% (Table 4). Swan et al.'s (2021) model's high performance may be due to an exhaustive use of the full training data, which consists of 35K images compared to 1.3 K images used by the models of this study. The correlation of the crowdsourced dataset size on test performance may also be signified by the U-Net and FCN models' higher f1 scores when trained on the default training set size compared to a smaller training set size (Table 2). Furthermore, the poor performance on *Big Rock* classifications may be due to the low proportion of *Big Rock* labels in the AI4Mars dataset, which is 1.97% compared to the second smallest

category *sand* at 15.61% (Swan et al., 2021). If we assume a comparable distribution in our training set of 1.3K images, there would only be ~26 labels for big rocks, which may be too insignificant for our models to learn.

Overall, given the fact that the AI4Mars project was crowdsourced through inexperienced non-experts and the relatively restricted size of the training set, the performances of the proposed models in this study were acceptable. Future studies could look into training more recent state-of-art models, such as the InternImage model (Wang et al., 2022), on this dataset.

Code Availabilities

All code used to produce the results of this study is included in a Jupyter Notebook on the Kaggle Platform: https://www.kaggle.com/code/zivvvo/ai4mars-comp-596

Furthermore, saved model weights to generate these results are also included in the project submission. Contact zhe.fan@mail.mcgill.ca for additional inquiries.

References

Alexandre, M. (2023). *Pytorch-UNet: Pytorch implementation of the U-Net for image semantic segmentation with high quality images*. GitHub. Retrieved April 27, 2023, from https://github.com/milesial/Pytorch-UNet

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431-3440).

PyTorch. (2017). *FCN*. FCN - Torchvision main documentation. Retrieved April 27, 2023, from https://pytorch.org/vision/main/models/fcn.html

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18* (pp. 234-241). Springer International Publishing.

Swan, R. M., Atha, D., Leopold, H. A., Gildner, M., Oij, S., Chiu, C., & Ono, M. (2021). Ai4mars: A dataset for terrain-aware autonomous driving on mars. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1982-1991).

Wang, W., Dai, J., Chen, Z., Huang, Z., Li, Z., Zhu, X., ... & Qiao, Y. (2022). Internimage: Exploring large-scale vision foundation models with deformable convolutions. *arXiv preprint arXiv:2211.05778*.

Appendix

Appendix A:
Validation Confusion Matrices (Raw Count) for U-Net 3-Fold Validation with default training size:
Fold 1:
[[3065378 547553 2303832   0]
 [274064 3889421   161466   0]
 [43799   81786 1086404   0]
 [286   30995   805   0]]

Fold 2:
[[3585400   915224 1125990   0]
 [605712 2647709   308391   0]
 [12826   117848   916405   0]
 [1487   14567   413   0]]

Fold 3:
[[5023305   247143   440291   0]
 [159719 3460344   43526   0]
 [56207   86495   654063   0]
 [369   11461   9   0]]

Validation Confusion Matrices (Raw Count) for FCN 3-Fold Validation with default training size:

Fold 1:
[[6605981   117162   17840   19]
 [350126 7735378   26052   5259]
 [218435   73048 1503362   2991]
 [3524   71676   6350   13887   ]]

Fold 2:
[[6070909   72648   48226   160]
 [246239 7285102   26250   2026]
 [67591   44287 1741196   1714]
 [2738   58013   1252   11767]]

Fold 3:
[[6204153   168299   26954   0]
 [55815 7896021   11275   2689]
 [44275   74830 1497465   1357]
 [2350   51171   2733   8366]]