CS230

Estimating Access to Mobile Broadband

Eric Colbert
Department of Mechanical Engineering
Stanford University
colber94@stanford.edu

Tyler Robbins
School of Law
Stanford University
tdrobbns@stanford.edu

1 Problem Description

Americans are disconnected. The FCC estimates about 22 million Americans today do not have access to broadband. [1] However, the FCC's broadband data are notoriously inaccurate and severely under-represent the number of Americans without access to the internet or with access to only one ISP. [2] Independently gathered data suggest the true number of Americans without broadband is closer to double the federal government's estimate. [3].

The FCC's data is so inaccurate because of its high granularity. An ISP that serves even person in a census block is counted as serving the entire census block. Gathering more accurate data is possible, but incredibly difficult. To create a wireless broadband map, for example, the California Public Utilities Commission had employees travel around the state to physically test where there is cell service. That data has not been updated since 2017.

Better broadband deployment estimates are essential to making well informed policy decisions, from how the government should allocate resources to justifications for net neutrality. Thus, we propose using deep learning techniques on satellite and geographic information systems data to predict access to broadband. Because the data is more available, we will focus our efforts on predicting access to mobile broadband.

2 Related Work

This problem comes down to image segmentation, namely dividing the image into various segments to understand the meaningful classes of objects. This is a classic problem for computer vision and we took a dive into these types of problems. A quick literature search turned up no projects that directly worked on this area, but some that dealt with image segmentation.

CNN's are very well suited for this type of problem and research was done to look into using deep neural networks such as ResNet [4], the state-of-the-art network that won the 2016 ImageNet competition and is widely used in semantic segmentation problems. Researching into this, we stumbled up another architecture, namely U-Net [5], that is a CNN originally developed for biomedical image segmentation. This model works well off of small training sets and yields precise segmentation's than other sliding-window CNNs [6].

This U-Net architecture utilizes a scheme that replaces pooling operators with up-sampling operators that increase the resolution of the output. Also, by combining the features from the up-sampled

[1] 2019 Broadband Deployment Report. https://docs.fcc.gov/public/attachments/FCC-19-44A1.pdf

[2] Drew Clark. Broadband Mapping Is a Mess. No One Knows What To Do About It. https://www.bbcmag.com/law-and-policy/broadband-mapping-is-a-mess-no-one-knows-what-to-do-about-it

[3] https://broadbandnow.com/research/fcc-underestimates-unserved-by-50-percent

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

with high resolution features of the contracting path, this model can perform localization. The down-sampling and up-sampling paths are more or less symmetric, hence a u-shaped architecture.

3 Data Set

Building upon the milestone 1, we continued to build on our own data set. This data set consists of satellite images and a ground truth segmentation mask representing what percentage of the population in an area has access to mobile broadband. We used publicly available satellite images taken by the European Union's Sentinel-2 satellite. To improve upon our results from Milestone 1, we added an additional channel to these images that marked locations of cell towers.

To create the ground truth segmentation masks, we used Mobile Deployment data from the FCC. To allow for different levels of complexity, we created three different representations of "access to broadband" with 2, 5, and 10 categories. Each category represents a percentage bin. For example, in the 5 category representation, the bins are 0-20%, 21-40%, 41-60%, 61-80%, and 81-100% access to broadband. For Milestone 1, we used the 2 category representation to quick experimentation and iteration on Google Colab. For Milestone 2, we switched to the 5 category representation and trained on AWS. For our final implemetation we continued with 5 classes and utilized AWS, Kaggle, and Google Colab to speed up testing.

Our original testing utilized purely California data because we thought the state's geographic diversity would lend well to training. We later added data from Colorado, Alaska, Oregon, New Mexico, Alabama, and Mississippi for even greater geographic and LTE coverage diversity.

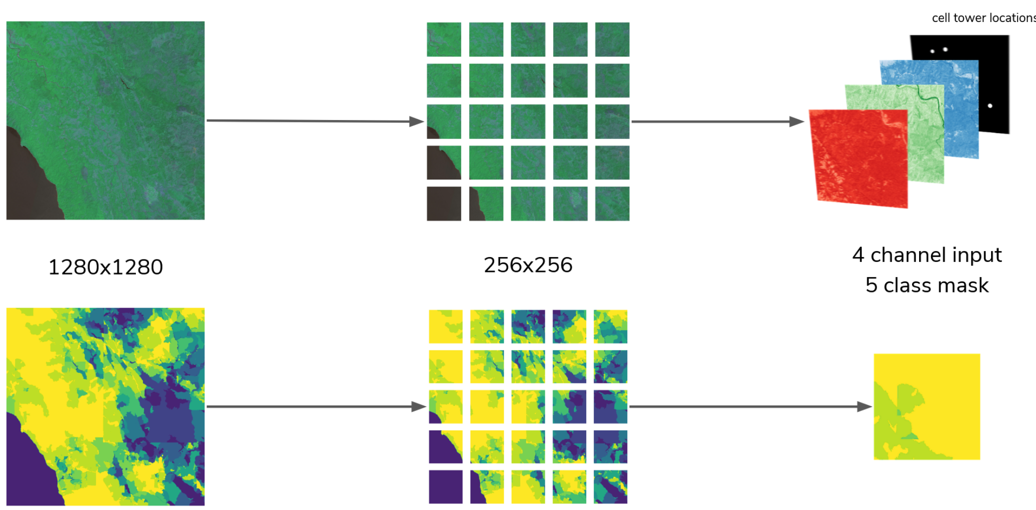

Figure 1: Data preprocessing pipeline

3.1 Preprocessing

The original satellite images were on the order of 5490x5490 pixels. To speed up training, we split the raw satellite images into various sized images, but found the greatest success with splitting these original into 1280x1280 images, then down-sampled those images to 256x256 pixels images for training. This preprocessing of the images allowed us to utilize data from more states, sped up training, and resulted in higher accuracy.

3.2 Data Augmentation

A good strategy to increase data and train a more robust model is data augmentation. We implemented a generator that can, in real time, augment data in a variety of ways such as cropping, rotating, mirroring, etc. This was added to our pipeline as a way to decrease over-fitting.

4 Method

We implemented to the aforementioned Unet model for our image segmentation problem. The network has 23 total convolutional layers, split into the contracting and up-sampling paths. The contracting path utilizes repeated 3x3 convolutions, followed by a ReLU unit, and finally a 2x2 max pooling with stride 2. The up-sampling consists of repeated layers of 2x2 up-convolutions, two 3x3 convolutions, and a ReLU unit. The final layer applies a 1x1 convolutional layer to map the feature vector to the desired number of classes.

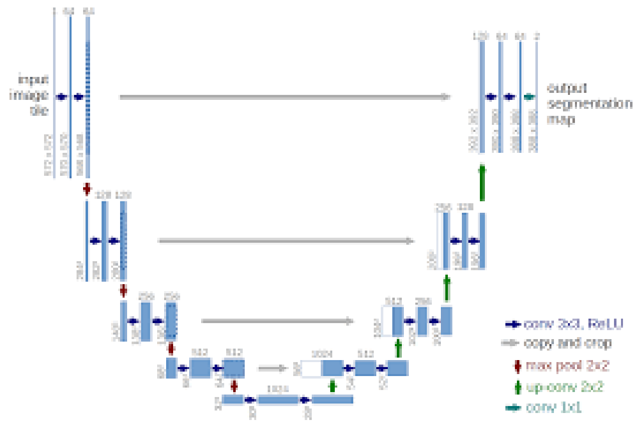

Figure 2: UNET Architecture

In a departure from standard implementations of the Unet model, we added batch normalization in the model directly after the activation in both the up-sampling and down-sampling.

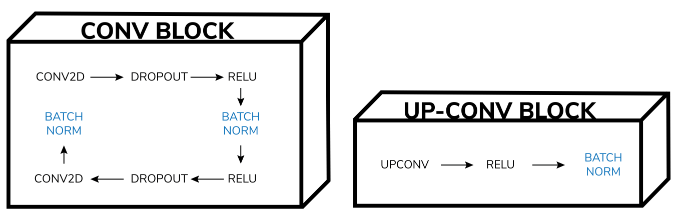

Figure 3: Batch Normalization added into U-Net Model

In our final model We use an Adam optimizer with a learning rate of .0001, and a batch size of 16. We split our training and dev set with an 80:20 ratio. We run our model for 500-1000 epochs. During our testing, we adjusted dropout, learning rate, input image size, and training epochs, number of root filters, and layer depth to see how these hyper-parameters affect performance. We evaluated model performance using F1 scores.

As our problem is a multi-class classification problem, our categorical cross-entropy loss function is as follows:

$$L(y, \hat{y}) = -\sum_{j=0}^{M}\sum_{i=0}^{N}(y_{ij} * log(\hat{y}_{ij})) \tag{1}$$

5 Results

With training on 256x256 images of the various states, using 5 classes, we were able to achieve around 97% categorical accuracy on training data and 80% on the dev set.

Adding batch normalization to the U-Net model significantly sped up the training and reduced the variance between the training and datasets, albeit not eliminating it.

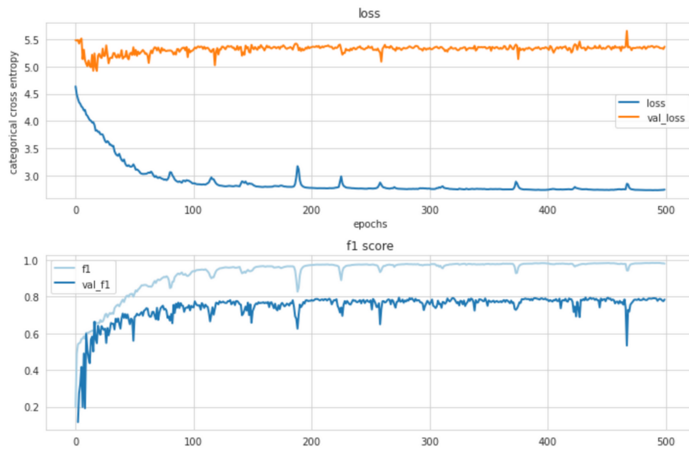

Figure 4: Loss function and F1 scores for training and validation.

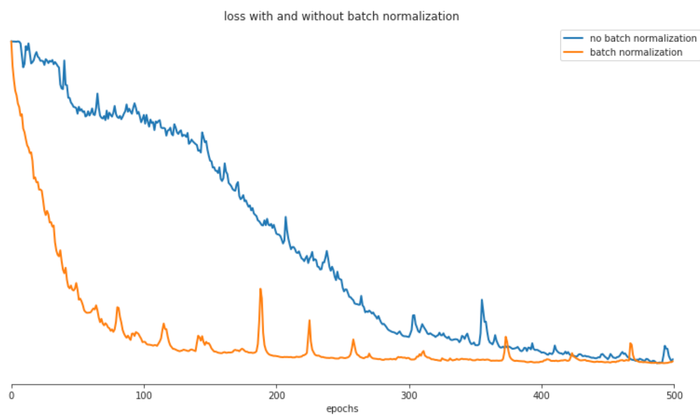

Figure 5: Loss during training on runs with and without batch normalization.

To test out how effectively this model is, we used images previously not seen before and generated predicted coverage mappings using the model parameters. These generated results, along with a reconstructed segmentation map from the training set, are shown in Figure 6.

Overall, the model creates accurate, high-fidelity predictions on the training set but still has issues in areas that have very mixed access in close proximity. Predictions on test set images show good mapping for large areas of consistent access, but really struggles with mixed access in close proximity. The process of splitting the satellite images into 256x256 chunks created artifacts in the predictions, which are especially visible on testing images.

6 Insights

We are convinced that this project can be improved by training it on more data across the country. Throughout development, we found adding data from states with different geographies and access profiles yielded most reduction in variance. Ultimately, we were too limited in time and VRAM to effectively and efficiently train on more data than what we used. We believe making the code more memory efficiency or just having access to more VRAM in training could help increase the model performance by allowing training on larger portions of, or even the full, satellite images at once. Chopping the images in smaller chunks creates artifacts in some situations, such as when a cell tower fell close to the edge of a dividing line.

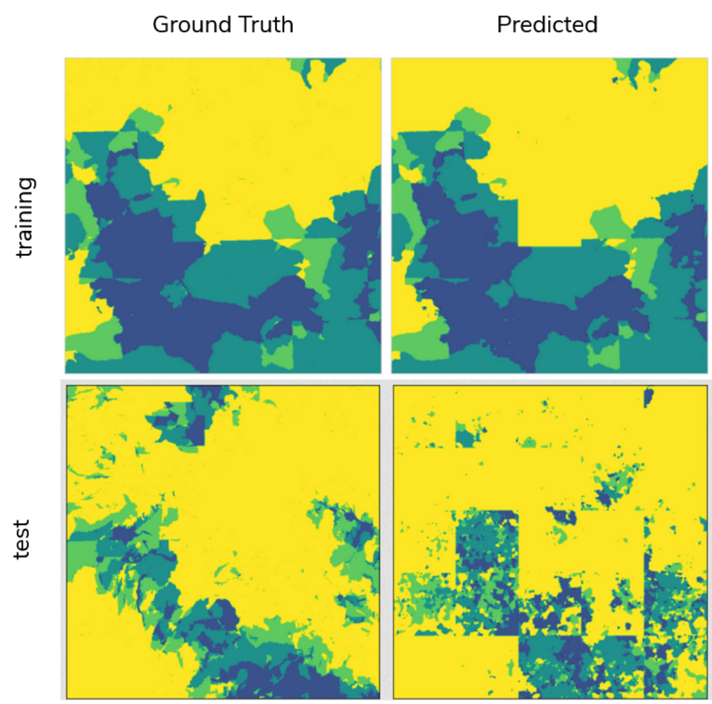

Figure 6: Example reconstructed segmentation masks from the training and test sets

However, as it currently stands we are confident this can be used by local and state governments, broadband providers, and independent organizations to accurately determine broadband access throughout the country. This model is robust, capable of combing a wide variety of geographical features and cell tower locations to reliably predict broadband access at a variety of levels. It can be an inexpensive and easily applied tool to help identify areas of need, find optimal placements for new cell towers, and, ultimately, ensure just and equitable access to mobile broadband throughout the country.

7 References

[1] 2019 Broadband Deployment Report. https://docs.fcc.gov/public/attachments/FCC-19-44A1.pdf

[2] Drew Clark. Broadband Mapping Is a Mess. No One Knows What To Do About It. https://www.bbcmag.com/law-and-policy/broadband-mapping-is-a-mess-no-one-knows-what-to-do-about-it

[3] https://broadbandnow.com/research/fcc-underestimates-unserved-by-50-percent

[4] He, Zhang, Ren, and Sun J. (2019) Deep Residual Learning for Image Recognition. [online] Available at: https://arxiv.org/abs/1512.03385

[5] Reonnebreger, Fischer, and Brox (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation [online] Available at: https://arxiv.org/pdf/1505.04597.pdf

[6] Ciresan, D.C., Gambardella, L.M., Giusti, A., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: NIPS. pp.2852–2860 (2012)