FINAL PROJECT REPORT

GMMA 847

BRANDON TOUSHAN, 10178340

MARCH 14, 2021

A framework for predicting poverty using satellite images and machine learning (Ethiopia 2017-2018)

EXECUTIVE SUMMARY

The proliferation of satellite imagery and open-source data into the public domain has the potential to drastically change humanitarian operations in the coming years. Computer vision and machine learning techniques have empowered practitioners with the ability to analyze millions of images from remote sources almost instantaneously. These new capabilities have opened the door to more insights from the ground being available than ever before. By tapping into these insights, humanitarian organizations are able to act upon far more up-to-date data when they make critical, time-sensitive decisions, resulting in better allocated resources, more successful response operations and more lives saved.

This project sought to replicate and build upon the work of Jean et al. and Johnathan Whitaker, and many others in building, testing and evaluating a framework for building easily reproducible machine learning algorithms for predicting poverty using satellite imagery. Socioeconomic data was obtained from the Central Statistical Agency of Ethiopia via the World Bank and satellite images were obtained from Google Earth Engine and Google Static Maps.

A pre-built Convolutional Neural Network (CNN) Architecture from fast.ai was fitted to the image data, with the last fully connected layer of the neural network used a feature extractor. A highly-tuned random forest model was then fitted to the extracted features and used to predict consumption on a household-by-household basis. The resulting predictor was evaluated in similar manner to Jean et al. and Whitaker (0.060 R2 Score) and demonstrated the framework's ability to produce a working model, despite serious data limitations.

INTRODUCTION

In the humanitarian world, change can happen all at once. Disasters strike, populations move and economic activities change. Once prosperous regions can be become poverty-stricken and in dire need of aid almost overnight. At the moment, the best that a humanitarian organization can do is respond to these crises based on the most up-to-date data they have from the ground. Problems arise when this data is upwards of a year old (as is common with census and survey data, the gold standard for population research) leading to all manner of potentially disastrous problems during relief efforts (poorly allocated/targeted resources, logistical problems, etc.).

Utilizing personnel on the ground is one such solution to this problem, albeit an extremely costly one with serious limitations to boot (one cannot be everywhere at once). Enter satellite imagery. Advances in remote sensing technology and the proliferation of satellite imagery into the public domain has opened up a brand new channel for monitoring crises the world over. Imagery providers, such as Planet and Google Earth (which was used to develop this project), are now capable of providing extremely high resolution images of the earth via almost real-time data feeds. These advances have provided humanitarian organizations with the ability to monitor developing crises anywhere in the world with the click of a button. Unfortunately, despite all their promise, these advances are held back by the very same personnel constraints that plague more primitive solutions.

Analyzing satellite imagery is an extremely difficult task for humans. It takes a tremendous amount of time to train a person to be able to identify indicators of poverty from aerial images. It takes even longer still to perform said analysis, especially on a comparative image-to-image basis. Given the considerably massive quantity of images that need to be analyzed in order for useful insights to be extracted, an extremely difficult task quickly becomes an impossible one. Enter machine learning and computer vision.

By teaching a computer (more specifically an algorithm) how to “see” and analyze satellite images, the personnel constraint is removed. Processing constraints aside, machine learning and computer vision techniques (which will be covered in detail in later sections) can be used to analyze any number of images extraordinarily quickly and derive insights almost as fast as images are procured. Machine learning techniques provide the added benefit of fully quantifying desired variables on an empirical basis, providing humanitarian organizations with the ability to predict and forecast metrics, such as household consumption and unemployment, using nothing more than the images themselves.

This project has has set out to evaluate, recreate and build upon the work of Jean et al., Yeh et al., Johnathan Whitaker and many others on the use of machine learning and computer vision to predict household consumption (and hence poverty) using solely satellite image data. Ethiopia was selected as the specific use case for this project, given its similarities to Malawi (which has been used extensively in literature), the availability of relatively up-to-date data (2019 census data that just recently became publicly available via the World Bank) and the Canadian Red Cross’s connection to humanitarian operations in the country via their five-year initiative with the Ethiopian Red Cross Society.

BACKGROUND

Using machine learning and satellite imagery to predict poverty is nothing new. Jean et al.’s 2016 paper (which was later reproduced using the modern software tools used in this project by Johnathan Whitaker), *Combining satellite imagery and machine learning to predict poverty,* produced outstanding results predicting poverty in 2016-2017 Malawi using a combination of satellite imagery (daytime and nighttime lights) and household consumption data. Yeh et al.’s 2020 work, *Using publicly available satellite imagery and deep* *learning to understand economic well-being in Africa*, produced similarly outstanding results using satellite imagery (daytime and nighttime lights) and household asset data to predict poverty in 2018-2019 Nigeria and greater Africa. These two papers (and the work of Whitaker transcribing Jean et al.'s paper into open-source code) represent the absolute state-of-the-art in the field and the basis for this project.

Both papers (and Whitaker's adaption of Jean et al.'s 2016 paper) made use of convolutional neural networks (CNN) to model satellite imagery data and create predictions. The key difference between Jean et al.'s and Yeh et al.'s work is the use of nighttime light intensities as intermediate labels for training a feature extractor on daytime imagery (as was done in this project). Yeh et al. instead chose to incorporate both sets of imagery (daytime and nighttime) in the model end-to-end, with models for each trained separately and then joined in a final fully connected layer at the end of the neural network. As a result of this method, Yeh et al. were able to train a model that learned features in both daytime and nighttime imagery without prescribing what features the model should look for beforehand (unlike Jean et al.'s method wherein nighttime images were used to prescribe what should be looked for in daytime images via feature extraction).

As the end goal of the project was to develop and evaluate the potential of a generic modelling framework for Red Cross use in humanitarian emergencies, modelling decisions prioritized simplicity, reproducibility and ease-of-use. With this in mind, Jean et al.'s feature extractor method was preferred over Yet et al.'s end-to-end method, a pre-trained neural net from fast.ai was selected in lieu of a custom build and a random forest model with an easily reproducible hyper parameter tuning process was developed.

For industrial purposes (such as those of the Red Cross), the benefits these choices provide relative to the more sophisticated (or less complete, in the case of Whitaker) work of Jean et al. and Yeh et al. is immense. Firstly, the feature extractor method used is far easier to reproduce in a production environment and allows for the bulk of the modelling process to be broken down into smaller more digestible chunks, should computing resources prove a constraint. Secondly, the use of a pre-trained neural net places far less burden on the end-user in terms of domain knowledge requirements and saves a considerable amount of time at the model construction stage. Finally, the developed process for hyper parameter tuning empowers the end user with the ability to build and tune a bespoke, working, machine learning model for their specific data set, without requiring a considerable machine learning background (going beyond Jean et al. and Whitaker's use of less accessible ML techniques).

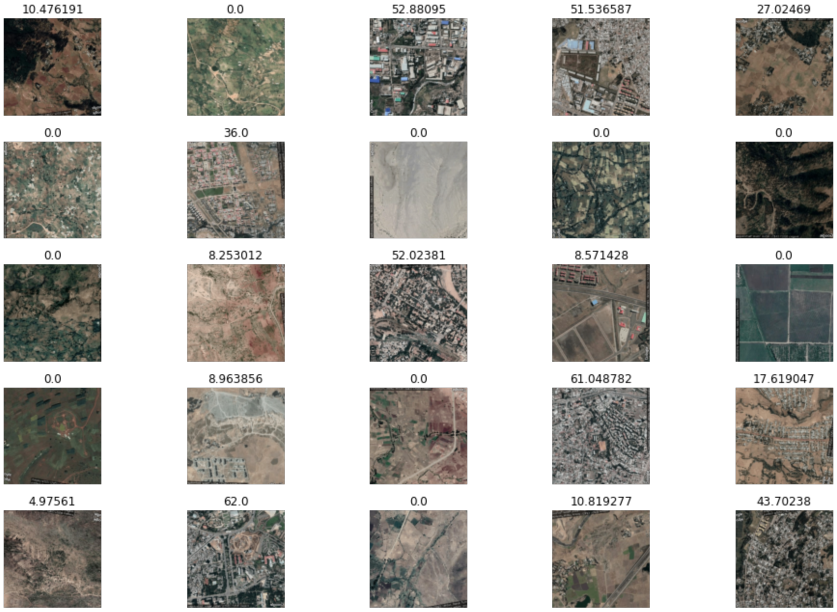
The combination of these three elements provides the end user (in this case the Red Cross) with a framework for constructing a working model for any country with Geo-tagged consumption/asset data, that is relatively simple to use and still enjoys performance comparable to much more manual, state-of-the-art techniques. Applied to data from Ethiopia's 2018-2019 census (likely the first project to model this newly available data set in this way), this framework was able to produce and tune a working algorithm for predicting consumption, despite vastly inferior data quality and quantity than the 2016 Malawi data used by Jean et al. and Whitaker.

IMPLEMENTATION

Applying the framework to a new dataset (in this case the Central Statistical Agency of Ethiopia's 2018-2019 Socioeconomic Survey) takes place in several distinct phases. Data needs to be loaded and cleaned, corresponding images (night and daytime) need to be obtained and processed for modelling, the pre-built neural net needs to be loaded from fast.ai and trained on the data, features need to be extracted from the model and transformed, and a random forest model needs to be fit to the extracted features and tuned in order to predict consumption. End-to-end, the entire process (including the time required to download ~5k images from Google Static Maps and Google Earth Engine) took just under 4 hours to run on a Google Collab Pro virtual machine using a Nvidia Tesla T4 16GB GPU with 25GBs of RAM, demonstrating the rapidness of the framework and its ability to be run on an extremely affordable virtual machine (Google Collab Pro costs roughly $10/month).

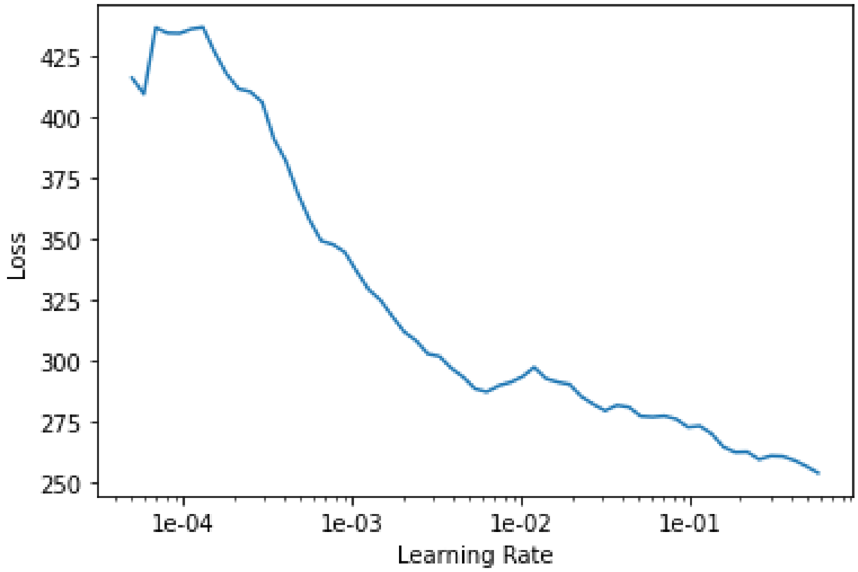
For the socioeconomic data component, the following variables need to be obtained at a household level; a poverty metric (consumption, asset wealth, etc.), a household survey weighting factor, an urban/rural classifier and the latitude/longitude of each surveyed household (these are typically obscured somewhat in publicly available datasets). Individual household data also needs to be clustered into aggregate groups to account for the best resolution of satellite imagery available to the end user (roughly 10km in the case of Google Earth Engine). Once cluster data has been obtained and processed, nighttime and daytime imagery needs to be obtained (Google Static Maps and Google Earth Engine were used for this purpose) for each of the corresponding cluster coordinates (one image of each type per cluster) and processed for modelling (lighting, zoom, tint, size, etc.).

Figure 1: Image Modelling Epoch



For the modelling component, a convolutional neural network (CNN) needs to be constructed to model the satellite imagery (for this project a pre-built architecture from fast.ai was chosen to maximize ease-of-use and reproducibility). Without delving too deeply into the architecture, a CNN is needed due to this network's unique ability to automatically detect important features in the data without any human intervention, drastically reducing the amount of user input and domain knowledge required to build a bespoke, working classifier. Once the network has been constructed (or in this case loaded), a loss function needs to be defined (MSE was selected for this project given the ratio nature of the dataset), a variety of learning rates need to be tested (in order to find the optimal learning rate for fitting the data) and the algorithm needs to be fit to the data and cross-validated (due to the tendency of CNN's to over-fit).

Figure 2: Loss vs. Learning Rate



Once it has been fit to the data, the final fully connected layer of the neural network needs to be extracted and used to create a dictionary of features for predicting poverty (in this case consumption). These features need to be assembled into a data frame and processed into a final training dataset (cleaned, scaled and split). The final training data then needs to be fit to a random forest classifier (which resulted in drastically better performance than any other algorithm), which needs to be hyperparameter tuned (in this project an easily repeatable method utilizing both random and grid searches was implemented) and then used to make predictions. The resulting predictions need to be evaluated using the testing dataset (R2 score and Mean Absolute Error were chosen for their academic and in-the-field significance), after which the model is deemed either good enough to enter production or discarded.

RESULTS

Two random forest models were trained in the end; one using the extracted features from the CNN model and one using nighttime lights data only. After extensive hyperparameter tuning, the extracted features model produced a R2 Score of 0.060 and a Mean Absolute Error of 17.87, while the nightlights only model produced a R2 Score of -0.072 and a Mean Absolute Error of 20.74. Logistic regression and ridge models were also tested; however, these models failed to compete with the performance of the random forest models on a like-to-like basis.

Evidently, these scores are dramatically lower than those of Jean et al. and Whitaker, which can be attributed to poorer quality and less numerous data than what has been used in literature. The Malawi 2016 dataset used by Jean et al. and Whitaker possessed upwards of four [4] times as many households as were present in the 2018-2019 Ethiopia dataset used for this project and was significantly cleaner. The Ethiopian dataset was also much more geographically diverse, further complicating the modelling process and leading to a less accurate predictor in the end.

While somewhat disappointing, these results serve to validate the methodology of the modelling framework as a whole. Despite extremely lacklustre data (that was not able to produce a working model with nighttime lights alone, unlike what was observed by Jean et al. and Whitaker). The feature extraction and hyperparameter tuning techniques used in the modelling process managed to “save” the model, despite poor data quality, demonstrating immense promise for the modelling framework itself. Given these results, it is not a massive stretch to believe that this modelling framework may well be applied successfully to just about any geographic dataset with similar inputs.

CONCLUSION

Despite the limitations of the dataset, a working predictor of poverty in Ethiopia was developed using the extracted features model. It may be quite inaccurate (MAE of 17.87), but it still managed to identify a signal after extensive hyperparameter tuning (R2 score of 0.060), something that was impossible to accomplish with solely nighttime lights (R2 score of -0.072). Given the poor quality of the dataset used, this result is nothing short of extraordinary and serves as a great validation of the modelling framework that has been created. Even the worst quality data, from a country with a difficult geography to model, resulted in a working model with some use as a baseline predictor of poverty on the ground.

For the Canadian Red Cross, this framework provides a way to obtain a baseline understanding of exactly what is happening on the ground anywhere in the world, at any time. As was demonstrated with Ethiopia (with a poor dataset no less), this framework is capable of producing a machine learning model that can predict poverty at a household level using nothing more than satellite images and socioeconomic data. Thanks to the outstanding work of Jean et al., Yeh et al. and Johnathan Whitaker, a foundation has been laid for a simple, easy-to-use tool with the potential to aid in humanitarian operations for the foreseeable future.

Where this tool goes from here is entirely at the discretion of the Canadian Red Cross. Effective use of this framework has already been demonstrated in Malawi (Jean et al. and Whitaker) and Ethiopia (this project); however, the potential applications are endless. Thanks to the uniquely reproducible nature of the framework, anywhere in the world with the required data (daytime and nighttime satellite imagery, geotagged household data and some form of poverty metric) is fair game. With Canadian Red Cross operations currently well underway in Syria, Myanamr and Western Canada, logical starting points and use cases abound.

WORKS CITED

[1] Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. Remote Sensing of Environment.

[2] Central Statistical Agency of Ethiopia. Ethiopia Socioeconomic Survey (ESS4) 2018-2019. Public Use Dataset. Ref: ETH_2018_ESS_v01. Downloaded from https://microdata.worldbank.org/index.php/catalog/3823 on March 1, 2021.

[3] Jean, N., Burke, M., Xie, M., Davis, W. M., Lobell, D.B., and Ermon, S., 2016. Combining satellite imagery and machine learning to predict poverty. Science 353 (6301), 790-794.

[4] Yeh, C., Perez, A., Driscoll, A. *et al.* Using publicly available satellite imagery and deep learning to understand economic well-being in Africa. *Nat Commun* 11, 2583 (2020). https://doi.org/10.1038/s41467-020-16185-w

[5] Whitaker, J. (2019, November 18). Deep learning + remote SENSING – USING NNs to turn imagery into meaningful features. Retrieved March 12, 2021, from https://datasciencecastnet.home.blog/2019/11/12/deep-learning-remote-sensing-using-nns-to-turn-imagery-into-meaningful-features/

APPENDIX

See attached Jupyter notebook .pdf for source code, reproduction instructions and technical commentary