Machine Learning Engineer Nanodegree Capstone Proposal

Patrick Albert Mikoaski de Souza

April 4th, 2021

Detecting presence of fire in images (from Kaggle[1] datasets)

Field: Computer Vision

1. Domain Background

According to the wikipedia[2], **computer vision** is an interdisciplinary scientific field that deals with how computers can gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to understand and automate tasks that the human visual system can do. Computer vision tasks include methods for acquiring, processing, analyzing and understanding digital images, and extraction of high-dimensional data from the real world in order to produce numerical or symbolic information in the forms of decisions.

As a person who lives in Brazil, which is a country that suffers a lot from forest fires annually, I believe it is possible to facilitate detection and combat this type of problem using computer vision as a tool.

From a local perspective, a greepeace[3] article stated that the fires in 2020 are the worst in the last decade.

Early detection systems using computer vision with AI could help firefighters combat blazes, aid in recovery, and prevent wildfires from starting to begin with. In addition, it is possible that environmental and political advocates create control and prevention systems with the mapped data obtained.

2. Problem Statement

Accordly to Kaggle[1] dataset page, the data is divided in images that contain fire and regular images (non-fire images), so the problem is essencially a binary classification problem.

An image must be passed to the model, be processed and return if the image contains fire or not.

The plan is to use Convolutional Neural Networks and Transfer Learning to build the better model possible to detect fire presence into the images.

With cameras installed around the world, it would be possible to check critical points and susceptible to fire.

This model also can be implemented as a web app so the users can upload a image and verify if the image contain fire or not.

3. Datasets and inputs

The dataset is provided by a kaggle dataset page, which was created by a group which modelled a similar problem in Nasa Space Apps Challenge in 2018.

Data[1] is dividided in **fireimages** and **non-fireimages**. The fireimages contain images with fire and heavy smokes and the non-fireimages contain nature images in general.

The data is skewed, so some proper tratament or alternative metric need to be settled in the images analysis.

4. Solution Statement

The solution will be to construct a Convolutional Neural Network model which can rotulated images if they have fire or not, extracting common features.

Another good solution will be uses transfer learning to use already trained models to better feature extractions getting better results.

During the modeling phase, part of the data will be used to test the effectiveness of the model. However, for final results, I will get random images on the internet to test the effectiveness of the final model.

5. Benchmark Model

In the Kaggle page there are some models[4] made by the community for this project. One of them have 95% of accuracy using a convolutional neural network. My goal is to have – at least - this same level of accuracy and improve secondary metrics for better results.

6. Evaluation Metrics

In addition to the standard accuracy, another key metric to this solution is to minimize the false negatives, which generates the worst scenarios possible, when the model doesnt detect the fire presence even if the fire exist. So f1-score, precision, recall, confusion matrix will also be monitored.

7. Project Design

In addition to the traditional modeling process, I will implement the steps learned during the Machine Learning Engineering Nanodegree course, applying the AWS Sagemaker[5] service to build, train and produce this model.

Step 1: Data Processing

1. Create a notebook instance in Sagemaker
2. Download and prepare the data in the notebook
3. Verify the data characteristics inside of the notebook

Step 2: Data Modeling

4. Build the model
   (In this case, a keras/tf model using train.py)
5. Implement estimator and train the model

Step 3: Evaluate and Inference

6. Evaluate the performance compare different models (at least a common cnn model x transfer learning model)
7. Deploy the best model

8. Reference

[1] Kaggle data: https://www.kaggle.com/phylake1337/fire-dataset

[2] https://en.wikipedia.org/wiki/Computer_vision

[3] greenpeace.org/international/story/45383/brazil-fire-amazon-pantanal-bolsonaro/

[4] https://www.kaggle.com/chemamasamuel/fire-detection-96-accuracy

[5] https://aws.amazon.com/pt/getting-started/hands-on/build-train-deploy-machine-learning-model-sagemaker/