Validating a Google TensorFlow Image Classifier

Presented by: Bob LoGalbo, Leidos

October 2018

leidos

Conflation using TensorFlow

- Utilities do not always have asset inventory either completely recorded or completely described.
- Conflating is attempting to fully complete all descriptive aspects of an asset, including taking a full inventory of assets.
- Google TensorFlow can identify assets on images.
  - Images are a potential source of information which may help complete the inventory record for a type of asset or conflate a specific asset's record.

Google Street View
A Potential Source for Conflation Data

- Google Street View and potentially other sources of data can be sources from which asset records can be conflated.
- This presentation will discuss a method of evaluating how well TensorFlow behaves primarily as a function of Google Street View images as training/test material.

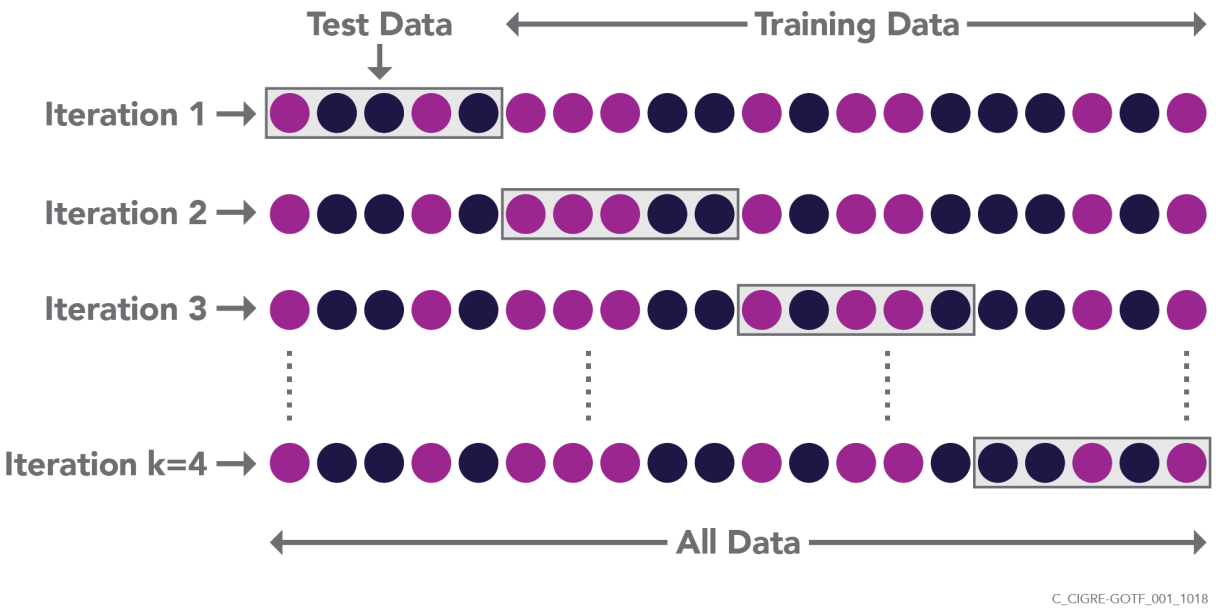
K-Fold Overview

- Create X bins, X = 2^(L-1) where L is the number of classes including the null class (i.e., E=empty class).
- Segregate GT images by type of class and multiclass combinations of appearance possible in any image, into the appropriate X bin.
- Apply stratified K-Fold training/test set selection where each of the X bins is considered its own “class” for the sake of stratified K-Fold validation. However, TP/FN/FP scoring shall only apply to each of the L classes within the given stratified layer.
- TNs shall not be counted, as they are defined as any object in a pic which is not of a desired class (a number that is ignored).

Why Conduct K-Fold Cross-Validation?

- It is important to segregate images with class-combinations in distinct sets.
  - Potential tendency for a classifier to more likely create false decisions for pictures which include combinations of classes
  - Example: likelihood *increases* for transformer classification errors when there are more than one type of class concurrently present in the same pic with the transformer

Brief Overview of K-Folds
Assuming K=4 with 20 Pics



- In this example, there are only 20 data samples and 4 folds.
- Each data point is then in a training set 3 times and test set once for a total of 4 tests to evaluate the efficacy of the classification method.

K-Fold Validation
Explained by Example

- First, divide all of the pics up by the number of different types of assets in each pic
- Then, assign the pic to a specific bin which only contains pics with that given combination of assets
- Example: If there are 5 types of assets, there are 2^5 bins.

K-Fold Validation
Explained by Example

If there are…

a. Capacitors
b. Fuses
c. Transformers
d. Banks
e. Streetlights

…there would be 2^5 bins (i.e. 32 bins)

	Asset Type				
Bin	a	b	c	d	e
0					
1					1
2				1	
3				1	1
4			1		
5			1		1
6			1	1	
7			1	1	1
8		1			
9		1			1
10		1		1	
11		1		1	1
12		1	1		
13		1	1		1
14		1	1	1	
15		1	1	1	1
16	1				
17	1				1
18	1			1	
19	1			1	1
20	1		1		
21	1		1		1
22	1		1	1	
23	1		1	1	1
24	1	1			
25	1	1			1
26	1	1		1	
27	1	1		1	1
28	1	1	1		
29	1	1	1		1
30	1	1	1	1	
31	1	1	1	1	1

K-Fold Validation
Explained by Example

- **Column I** is the total number pics available per bin.
- Number of pics for each of the 10 training folds is 90% of **Column 1**.
- Each pic is part of the training set 9 times out of 10.
- Each pic is part of the test set 1 out of 10 times.
- 10% of the pics at any one time make up the testing set.
- The TP/FN/FPs are measured uniquely per bin.
- Using Bin 6 as an example, each of the 10 folds measures the TPs/FPs/FNs of only **Asset Type C** and **Asset Type D**.
- Continuing with Bin 6 as the example, TensorFlow has a unique response when attempting to identify **Asset Type C** when **Asset Type D** is also present and vice-versa (hence the stratification).
- The total number of all TPs/FPs/FNs for all folds, for all assets, are totaled to evaluate metrics such as recall, precision, etc.

TP - True Positive FN - False Negative
FP - False Positive

Bin	Asset Type					Num Pics (Col I)	Num Training Pics Per Fold, K = 10 (Col II)	Num Testing Pics Per Fold, K = 10 (Col III)
	a	b	c	d	e			
0						250000	225000	25000
1					1	100000	90000	10000
2				1		100000	90000	10000
3				1	1	50000	45000	5000
4			1			100000	90000	10000
5			1		1	50000	45000	5000
6			1	1		50000	45000	5000
7			1	1	1	330000	297000	33000
8		1				100000	90000	10000
9		1			1	50000	45000	5000
10		1		1		50000	45000	5000
11		1		1	1	33000	29700	3300
12		1	1			50000	45000	5000
13		1	1		1	33000	29700	3300
14		1	1	1		33000	29700	3300
15		1	1	1	1	25000	22500	2500
16	1					100000	90000	10000
17	1				1	50000	45000	5000
18	1			1		50000	45000	5000
19	1			1	1	33000	29700	3300
20	1		1			50000	45000	5000
21	1		1		1	33000	29700	3300
22	1		1	1		33000	29700	3300
23	1		1	1	1	25000	22500	2500
24	1	1				50000	45000	5000
25	1	1			1	33000	29700	3300
26	1	1		1		33000	29700	3300
27	1	1		1	1	25000	22500	2500
28	1	1	1			33000	29700	3300
29	1	1	1		1	25000	22500	2500
30	1	1	1	1		25000	22500	2500
31	1	1	1	1	1	20000	18000	2000

Data Science of Scoring

- The next slide shows how to score TP, FN, and FP for an example pic.
  - The GT bounding box is the bounding box physically drawn by a person around each transformer in every pic used for training, testing, and validation.
  - The TF bounding box is the bounding box drawn by TF around what TF thought was a transformer.
  - The IOU ratio of every GT bounding box and every TF bounding box is evaluated. So, for GT = 3 boxes and TF = 4 boxes, 12 IOUs are calculated.
  - The IOU is the intersection of the areas of the 2 boxes divided by the union of the areas of the same 2 boxes. See next slide for graphical depiction of IOUs, scoring, and numerical examples.
  - If the IOU is above threshold (for these tests, threshold = .5), a TP is scored.

TP - True Positive FN - False Negative FP - False Positive TF - TensorFlow
IOU - Intersection Over Union GT - Ground Truth

Data Science Scoring (continued)

GT_m= mth Ground Truth Transformer

GT_m

In this pic, M=3 GT bounding boxes

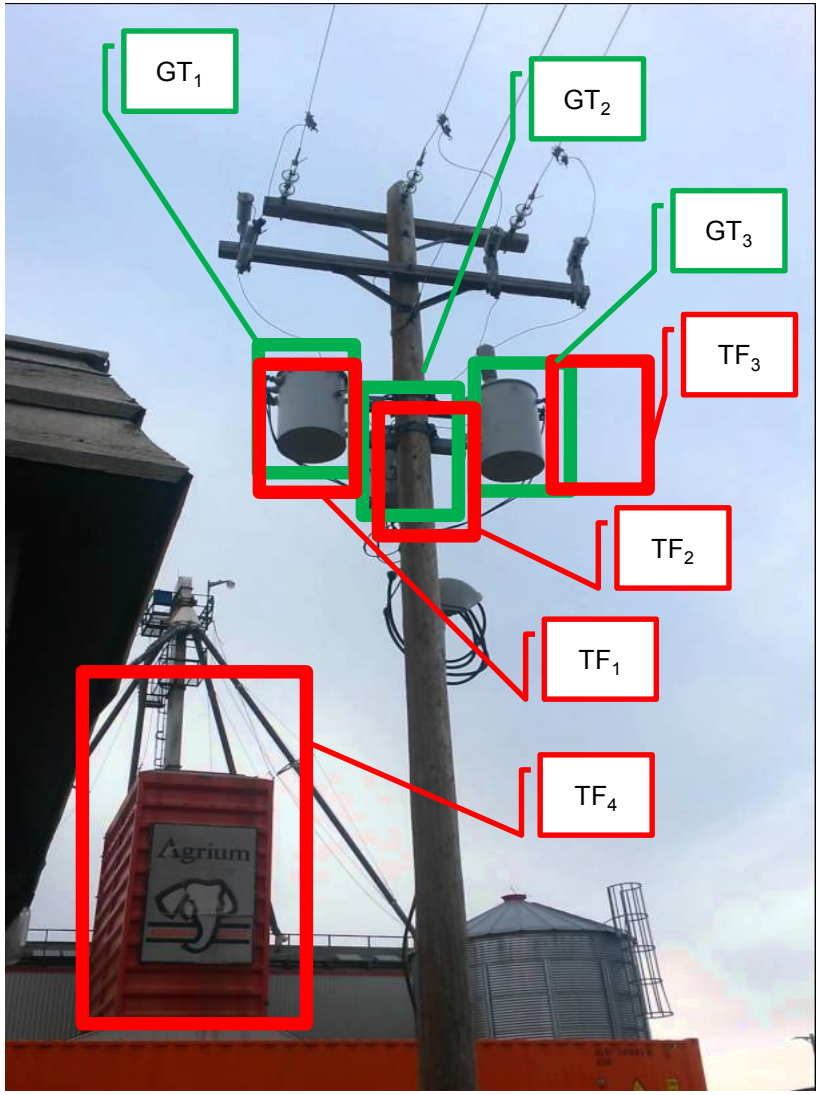
TF_n=nth TF found Transformer

TF_n

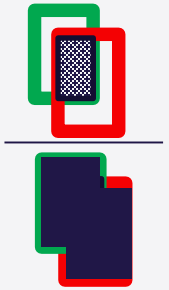
In this pic, N=4 TF bounding boxes

IOU_{mn} = IOU between GT_m and TF_n

KEY
GT = Ground Truth Bounding Box
TF = TensorFlow Bounding Boxes
TP = True Positive
FN = False Negative
FP = False Positive
IOU - Intersection Over Union
GT = TP + FN
N = TP + FP
Recall = TP/GT



Intersection Over Union (IOU)



Assume:
IOU_{11} = .85
IOU_{12} = 0
IOU_{13} = 0
IOU_{14} = 0
IOU_{21} = .05
IOU_{22} = .75
IOU_{23} = 0
IOU_{24} = 0
IOU_{31} = .5
IOU_{32} = .05
IOU_{33} = .18
IOU_{34} = 0

Total number of TPs = 2.
2 TF bounding boxes mapped to 2 GT bounding boxes (i.e. IOU_{11} > .5; IOU_{22} > .5) making TP =2.

Total number of FNs = 1.
1 GT bounding box not mapped to any TF bounding boxes (i.e. IOU_{3n} < .5 for n=1,2,3,4) making FN = 1.

Total number of FPs = 2.
2 TF bounding boxes not mapped to any GT bounding boxes (i.e. IOU_{m3} < .5 and IOU_{m4} < .5 for m=1,2,3) making FP = 2.

Summary

- We have described a method of testing how well conflation of images works using TensorFlow as a classifier and Google Streetview as a source of conflation data.
- The method of testing is K-Fold validation.
- We used stratification to isolate the effects of confounding influences when desired asset types appear in the same pic.

Thank You

Should you have questions, please contact:

Bob LoGalbo | Senior Chief Data Scientist, Leidos Engineering

phone: (407) 243-3875

Robert.D.LoGalbo@Leidos.com