Labelbox Consensus Calculation Method

John Thomas
August 6, 2019

1 Definitions

In this document I will use several different functions and variable names:

$$D_A \text{Datarow A in a Labelbox Dataset} \tag{1}$$

$$L_i^A : \text{Label i on Datarow } D_A \tag{2}$$

*(A Label in this context refers to the complete set of classificaitons and objects annotated on a given Datarow)

$$G_i^A : \text{Global Classification i on Label } L_A \tag{3}$$

$$O_i^A : \text{Object Class i on Label } L_A \tag{4}$$

$$P_i^A : \text{Polygon* i in class } O_A \tag{5}$$

*(For the purposes of this discussion, both freeform Polygons and Bounding Boxes are considered Polygons)

$$N_i^A : \text{Nested Classification i on shape } P_A \tag{6}$$

$$S_i^A : \text{Selection i on Classification } G_A \text{ or } N_A \tag{7}$$

$$C(A, B) : \text{The Consensus between two things A and B} \tag{8}$$

$$I(P_A, P_B) : \text{Intersection over Union between Polygons } P_A \text{ and } P_B \tag{9}$$

$$X_{total}^A : \text{the total number of Object } X^A \text{ ,where } X \in \{L, G, O, P, N, S\} \tag{10}$$

(11)

2 Classifications

Though this is covered in not insignificant detail in the Documentation, I will reiterate here, with more detail for the specific case of comparing only two labels.

Overall, any classification for which there is no corresponding selection on the other label will have the agreement for that specific class set to 0%, beyond that the exact method of computation depends on the type of classifier used:

2.1 Radio Selections

Radio Classification are the simplest, as they either agree, or they don't. When they are the same, the consensus between the two images for that class is simply 100% and when they disagree, it's 0%

$$C_{radio}(G_A, G_B) = \begin{cases} 0\% & S^A \neq S^B \\ 100\% & S^A = S^B \end{cases}$$

2.2 Checklists

Checklists operate off of a slightly different system. Checklists are calculated by the number of shared options divided by the number of options selected in one of the two labels, ie:

$$C_{checklist}(G_A, G_B) = 100 \cdot \frac{S^A \cap S^B}{max(S_{total}^A, S_{total}^B)}$$

This means that for a checklist with options $[1, 2, 3, 4]$, $C([1, 2, 3], [2, 3])$is the same as $C([1, 2, 3], [2, 3, 4])$

2.3 Text

Text is parsed exactly like a Radio Selection - if the two strings are exactly the same, then they return 100% consensus, however, if the are different then it returns 0. This is a simple string comparison, so it's case sensitive and will include any leading or trailing white space. For example: 'Test Text' / 'Test Text' will match, but 'test Text' / 'Test Text', or'Test Text ' / 'Test Text' will not.

2.4 Dropdown Classifiers

Dropdown Classifiers are saved the same as a Checklist in the label JSON: ["TopLevelOption", "NestedLevel1", "NestedLevel2"], and consensus is calculated on them in the same way. This means that if you have the Dropdown Options:

```
What is this image of:
Urban Area >
    Day
    Night
Rural Area >
    Day
    Night
```

Then $C(Urban > Day, Urban > Night) = C(Rural > Day, Urban > Day) = 50\%$, because they each share one option ("Day" and "Day" match with each other, regardless of the base option), and $max(S_{total}^A, S_{total}^B) = 2$ for both, because, despite only making one selection, there are two levels of nested classes. This means that, for dropdown classifiers, S_{total}^A represents not the number of selections made, but rather the total depth of the final selection.

3 Bounding Boxes and Polygons

The short version of how Bounding Box and Polygon consensus works is: Within each object class, each created shape is paired with a shape on the other label. This pairing happens in such a way as to maximize the total consensus calculation for each label, before accounting for nested classifications. Once the shapes are paired, the consensus between the two is calculated for each pair, and the average of all the scores is returned as the consensus for that object class.

3.1 Pairing Shapes

In order to pair shapes, it starts by pairing each shape off with the nearest shape on the other image. Then, if any shapes in image A point at the same shape in image B, it keeps the pair that is the best match, and re-pairs the other shapes against the remaining in Image B. The function repeats this process until each shape in image A has been paired with a unique shape in image B, or it has been left without a pair.

3.2 Geometry Agreement

Once the shapes are all paired, the actual agreement is calculated with and Intersection-over-Union calculation. Intersection over Union is a fairly standard agreement function in computer vision, and works by Dividing the total area covered by both shapes from the area shared by both shapes:

$$I(P_A, P_B) = \frac{P_A \cap P_B}{P_A \cup P_B}$$

In the case that a shape isn't paired with any other shape, the Agreement for that specific shape is set to 0.

3.3 Nested Classifications

The agreement for each nested classification is computed in the same method as the respective global classifier. However, the scope is obviously limited to only those classifications on the pair of shapes.

3.4 Pair Totals

From there, once a given pair has the agreement calculated for each respective nested class, the total agreement for the pair of shapes is set to the average of the Geometric agreement and each nested class. That is to say, the Geometry is given no more weight to the agreement between shapes than the classifications are. More formally, for a pair of shapes A and B, with nested classes N_i^A and N_i^B

$$C(P_A, P_B) = \frac{I(P_A, P_B) + \sum_i C(N_i^A, N_i^B)}{1 + max(N_{total}^A, N_{total}^B)}$$

3.5 Class Totals

Once the computation of each Pair's agreement has finished, a total Agreement is compiled for the whole Object Class. Simply, the total agreement for an object class is the average of the agreement of each set of pairs, with 0s included for any shapes that didn't have a pair.

$$C_{class}(O_A, O_B) = \frac{\sum_i C(P_i^A, P_i^B)}{max(P_{total}^A, P_{total}^B)}$$

4 Points and Lines

Although Labelbox supports point and line annotations, these annotations are not included in the Consensus calculation in any way, including any nested classifications within them.

5 Aggregation of Sub-Scores

It is important now to define what constitutes a single sub-score on a given image. Each global classifier is considered as its own Sub-Score on the image, however a geometric object class is taken as a whole when calculating the final agreement. Formally, a sub-score is one of the following:

$$C(G_1^A, G_1^B)$$
$$C(O_1^A, O_1^B)$$

Labelbox uses a simple average over each Sub-Score to calculate the final consensus of between two Labels on a given image.

$$C(L_A, L_B) = \frac{\sum_i C(G_i^A, G_i^B) + \sum_j C(O_j^A, O_j^B)}{max(G_{total}^A, G_{total}^B) + max(O_{total}^A, O_{total}^B)}$$

6 Final Label Consensus

To get the final Consensus Agreement that is displayed in the Labelbox Activity Log, The consensus is calculated between a given label, and every other label created on a that Datarow. These scores are then averaged, and the final Consensus Agreement is displayed on the Activity Log:

$$C_{final}(L_1^A) = \frac{\sum_{i/ne1} C(L_1^A, L_i^A)}{L_{total}^A - 1}$$

7 Benchmarks

The Benchmark Agreement is calculated in the exact same way, except there will only ever be one other label involved in the Agreement calculation for a given Labeler submission, the Benchmark Reference Label.