A Closed-Loop System for Improving Annotation Quality and Efficiency

Anonymous Author(s)
Affiliation
Address
email

Abstract

We present a general system and approach to improve the quality and efficiency of interactive annotation. A specific use case based on instance segmentation of vehicles for autonomous driving is used as an illustration. Via incremental AB testing and a custom analytics pipeline, we show how to optimize human-ML interaction to systematically improve annotation efficiency, and address the shortcomings of ML models.

1 Introduction

Pre-annotation and interactive (few-click) annotation are now commonly used in labelling tools and platforms to accelerate annotation and therefore reduce the cost of producing labelled datasets to train Deep Neural Networks (DNNs). In this paper, we present a general analytics-driven methodology that we have found useful to optimize our Machine Learning (ML) approach and incrementally improve its integration with human annotators.

2 Interactive annotation for segmentation

The annotation task considered in this study consists in labelling images to train computer vision models toward autonomous driving. Human annotators are required to draw accurate polygons around vehicles on those images. This is, however, a time-consuming and costly process.

Given the cost involved, multiple approaches have been suggested for machine-assisted instance segmentation. These usually consist of a deep learning-based segmentation of the object(s) integrated into a human-in-the-loop system. The human can interact with the system by correcting the model output, initializing the model with one or several clicks, or a combination of those steps. Examples of such systems include Polygon-RNN++ [1], DELSE [10], DEXTR [8], and CurveGCN [7].

We build upon an approach presented in [2] in order to illustrate our system. In a nutshell, this approach uses a few-click segmentation model (custom DEXTR) and a post-processing procedure that converts the produced raster mask into a sparse polygon. Figure 1 shows an example of human-edited ML output, compared with a fully manual approach, on the SYNTHIA-AL synthetic automotive dataset [12].

The goal is to conduct a series of AB tests in order to optimize the adequacy of the system's output for human editing, thus making the annotators as efficient as possible.

Submitted to 34th Conference on Neural Information Processing Systems (NeurIPS 2020). Do not distribute.

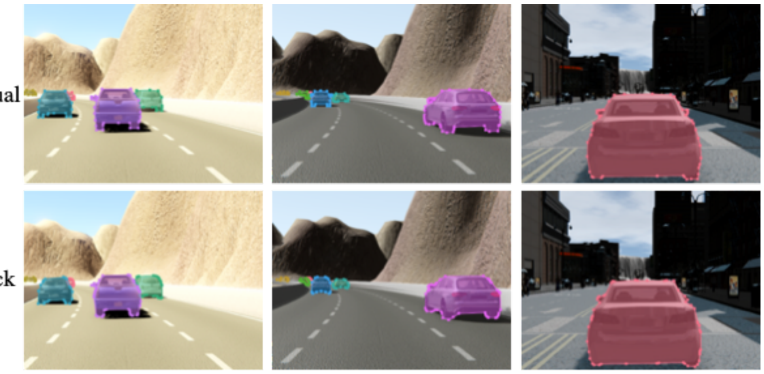


Figure 1: Comparing polygonal annotations of vehicles: (top) polygons drawn from scratch; (bottom) polygons manually edited from the output of the custom DEXTR.

3 Efficiency Metrics: a deep look into annotation analytics

3.1 Definition of the metrics

The performance of human annotators is dependent upon the quality of the interactions between the person and the labelling platform. As new features (such as ML-based automation) are introduced, it is of utmost importance to rigorously measure the impact of those features on annotation quality and efficiency. In order to achieve this, we have introduced a series of metrics that help us gather detailed insights into how annotation tasks are performed.

We have created a mapping between user tasks performed throughout the day, both on and off the annotation tool. We assign each type of activity that annotators do, on a daily basis, to a category and measure the amount of time devoted to them. The detailed categories are:

- **Training and feedback:** Before annotators begin work on a project, they go through a training program. Furthermore, once the project is in production, they receive constant feedback to improve their annotation work, based on the tasks performed so far.
- **Scanning:** Depending on the asset being labelled, finding the objects of interest can take a significant amount of time. This can include moving around a high-resolution image, with constant zooming in and out of regions, or navigating frames back and forth on a video.
- **Annotation creation:** This includes creating polygons that enclose objects of interest, in the case of vector annotation, as well a marking pixels in raster-based tasks.
- **Label selection:** Depending on the complexity of the task, finding the correct label to assign is not necessarily a quick operation. Because of that, we create a separate category to account for the amount of time spent searching for the right label, out of a catalog of available labels.
- **Annotation adjustment:** The process of initial polygon creation regularly requires annotators to adjust the shapes drawn. We track how much time is devoted to this type of task. In addition to this use case (i.e., create a shape and then adjust edges), there exists the possibility of asking annotators to redo a task, when the quality criteria are not met. It is expected that some of this feedback can be addressed by providing further adjustments to shapes.
- **Validation:** Annotators can perform a final check on their annotations, before they submit the work for a task to our platform. We want to assess how much of the time goes into this type of check.

For the case study laid out here we focus on two of the metrics, primarily: annotation creation and adjustment. For this, we have created a specific catalog of actions that can be performed on the user interface, which affect each of the categories. The tool reports back those actions into our analytics infrastructure, which allows us to create two data sources:

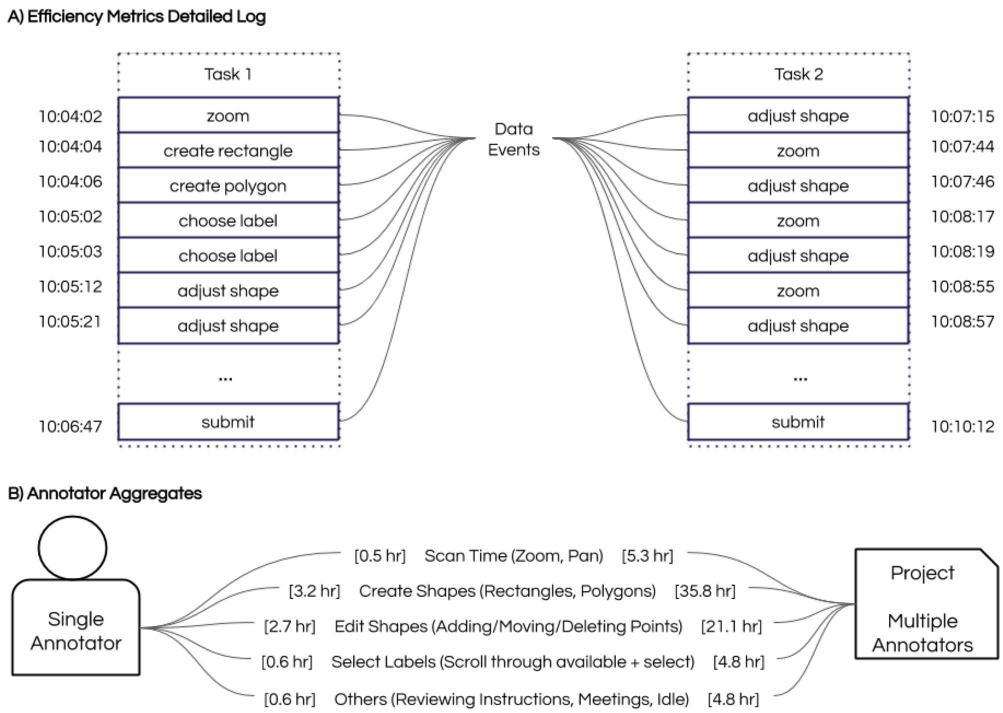
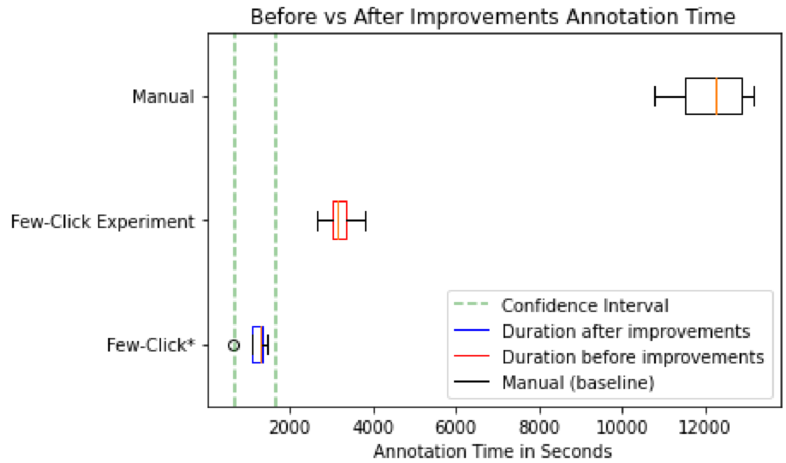


Figure 2: Efficiency Metrics data sources. **A)** The detailed log stores all events of interest processed by our internal data pipelines, which enables us to understand how each annotation task progressed. **B)** On top of the detailed data, we create daily aggregates per annotator, grouped by categories of interest, that are then fed into a statistical framework, for significance evaluation.

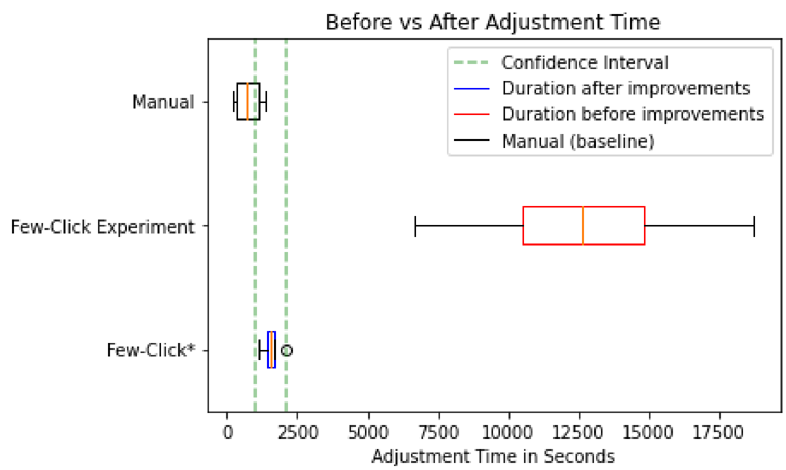
- **Efficiency metrics detailed log:** All user interface actions are stored sequentially for a given task so that we can reconstruct the actions performed on the annotation tool, as depicted in Figure 2.A. This is the finest granularity level in our analytics infrastructure and is useful to validate patterns. For example, one agent may choose to draw all shapes first and then do a second pass to select labels. Other agents may choose to select the label with each shape. Some other tasks may center around adjusting shapes after they are returned to annotators. That level of detail is only available through this source.

- **Annotator aggregates:** In order to power A/B testing, we need to define a granularity level to create *populations* that can be compared against each other. Figure 2.B shows two aggregation levels that we have found useful in our analyses. For instance, we may want to know if we reduced the daily number of hours that annotators spent creating polygons, due to some automation improvement. In that case we may define as *population individuals* the daily metrics gathered per annotator and then do either before/after or side by side comparisons between different groups. Creating this data source requires transforming individual actions from the detailed log into accurate time aggregates. We can then query population metrics based on the granularity of interest and assess the significance of the changes.

The combined usage of these two data sources becomes a critical asset in the way we can iterate quickly in our feature development. All of the metrics generated are updated on a near real-time basis, which allows members in different areas of the organization, from data scientists to project managers, to analyze the evolution of their projects and determine if the alleged improvements are verifiable.



(a) **Annotation Time**



(b) **Adjustment Time**

Figure 3: Comparison of average annotation time and adjustment time per object instance (in seconds) across all human annotators, for the baseline (*Manual*), the initial experiment (*Few-click*), and the experiment after improving the guidelines and the model output (*Few-click**).

3.2 A/B testing capabilities

In an effort to conduct evidence-based process and product development, we follow best practices in the industry [3, 5, 6, 9, 11] and adopt an experimentation approach that informs decision making. To this end, we have developed a flexible testing infrastructure that can consume data from multiple internal processes and is available to virtually anyone within the organization who wishes to setup and keep track of an experiment [5].

The adoption or rejection of software, UI/UX, and/or procedure changes that impact our annotation pipeline are evaluated through our AB testing framework, which measures the statistical impact of these changes on our efficiency metrics such as drawing and adjustment times. Significance of observed differences in a given efficiency metric is evaluated using rigorous statistical tests (*e.g.* two-tailed hypothesis test). For the duration of an AB experiment, treatments are randomly assigned and fixed to annotators from within our annotation tool [4].

4 Experimental setup

In the current experiment, we wish to compare the newly introduced interactive annotation system to a manual annotation baseline. Our setup is similar to the one described in [2]. In each group (A and B populations), each annotator is asked to label Motor Vehicles in 80 distinct images from a current customer project related to autonomous vehicles. Annotators are initially given clear guidelines along with a practice set, which is not counted towards in the metrics.

For this case study, efficiency metrics are gathered and stored for all annotators working on both treatments, and hypothesis tests evaluate two efficiency metrics of interest: initial polygon drawing time (from hereon dubbed annotation time) and adjustment time. We hypothesize that the new few-click tool will reduce both annotation and adjustment times.

5 Results and discussion

The first experiment did not yield the expected improvement (see *Manual* and *Few-click Experiment* in Figure 3). We confirmed our hypothesis concerning annotation time which saw a significant decrease, but had to reject the one pertaining to adjustment time, which was greatly increased. In order to understand what was happening, we selected images containing instances with very high adjustment time. After close inspection, we concluded the following:

- Our annotators entered sub-optimal extreme few-clicks which once fed into our custom DEXTR model along with corresponding images yielded low-quality polygons.
- Polygons on smaller objects had a much lower IoU on average when compared to established ground truths. We discovered that the source of this issue was in the parameters of the raster-to-polygon mask conversion algorithm.

We addressed the first issue by enhancing the training of our annotators and the labelling guidelines, and the second one by adjusting the parameters of the raster-to-polygon algorithm to maintain the quality of the output shapes at all scales. We then ran a new experiment whose results are shown as *Few-click** on Figure 3. Even though adjustment time remained slightly higher when using the few-click tool, the total labelling time decreased by 78%, a vast improvement over the initially observed 21% increase. Furthermore, the quality of the delivered polygons was maintained with an average IoU around 95% when compared to established ground truths.

6 Conclusion

In this paper, we described a general methodology to incrementally improve a human-in-the-loop system based on annotation analytics and AB testing, with a specific use case in image labelling for autonomous driving. We have shown that detailed analytics can help pinpoint shortcomings in annotation guidelines and improve the quality of the output of ML models.

Broader Impact

Our work is concerned with improving the efficiency of human-in-the-loop annotation systems. While ML is bound to automate certain tasks (including within annotation itself), we hope that optimizing human-ML interaction in annotation systems will direct human labour towards where it is needed, thus preserving the need for these jobs while removing the more tedious aspects of it.

References

[1] David Acuna, Huan Ling, Amlan Kar, and Sanja Fidler. Efficient annotation of segmentation datasets with polygon-rnn++. In *CVPR*, 2018.

[2] M. Bertrand, F. Ratle, and L. Juillard. Human-centric efficiency improvements in image annotation for autonomous driving. In *ICML Workshop on Human In the Loop Learning (HILL)*, 2020.

[3] Nikos Diamantopoulos, Jeffrey Wong, David Issa Mattos, Ilias Gerostathopoulos, Matthew Wardrop, Tobias Mao, and Colin McFarland. Engineering for a science-centric experimentation platform. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering in Practice*, ICSE-SEIP '20, page 191–200, New York, NY, USA, 2020. Association for Computing Machinery.

[4] Pavel Dmitriev, Brian Frasca, Somit Gupta, Ron Kohavi, and Garnet Vaz. Pitfalls of long-term online controlled experiments. In *Proceedings - 2016 IEEE International Conference on Big Data, Big Data 2016*, pages 1367–1376. Institute of Electrical and Electronics Engineers Inc., 2016.

[5] Raphael Lopez Kaufman, Jegar Pitchforth, and Lukas Vermeer. Democratizing online controlled experiments at booking.com. *CoRR*, abs/1710.08217, 2017.

[6] Ron Kohavi, Alex Deng, Brian Frasca, Toby Walker, Ya Xu, and Nils Pohlmann. Online controlled experiments at large scale. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume Part F128815, pages 1168–1176. Association for Computing Machinery, aug 2013.

[7] Huan Ling, Jun Gao, Amlan Kar, Wenzheng Chen, and Sanja Fidler. Fast interactive object annotation with curve-gcn. In *CVPR*, 2019.

[8] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool. Deep extreme cut: From extreme points to object segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[9] Diane Tang, Ashish Agarwal, Deirdre O'Brien, and Mike Meyer. Overlapping experiment infrastructure: More, better, faster experimentation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 17–26, 2010.

[10] Zian Wang, David Acuna, Huan Ling, Amlan Kar, and Sanja Fidler. Object instance annotation with deep extreme level set evolution. In *CVPR*, 2019.

[11] Huizhi Xie and Juliette Aurisset. Improving the sensitivity of online controlled experiments: Case studies at Netflix. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 13-17-August-2016, pages 645–654. Association for Computing Machinery, aug 2016.

[12] Javad Zolfaghari Bengar, Abel Gonzalez-Garcia, Gabriel Villalonga, Bogdan Raducanu, Hamed H Aghdam, Mikhail Mozerov, Antonio M Lopez, and Joost van de Weijer. Temporal coherence for active learning in videos. *arXiv preprint arXiv:1908.11757*, 2019.