Human-Debugging of Machines

Devi Parikh
Toyota Technological Institute at Chicago
dparikh@ttic.edu

C. Lawrence Zitnick
Microsoft Research (Redmond)
larryz@microsoft.com

Abstract

We have proposed the human-debugging paradigm, where human involvement is leveraged to identify bottlenecks in existing computational AI systems. We introduce this paradigm, and briefly describe two instances of our prior work on using this paradigm for computer vision tasks. We then describe some interesting challenges involved in employing this paradigm in practice.

1 Introduction

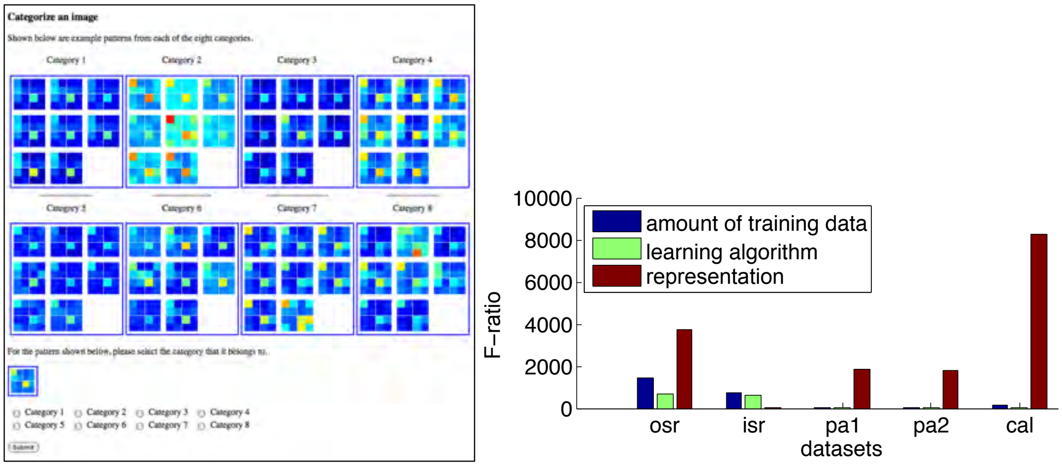
Improving the performance of computational AI systems offers many challenges. The community is constantly faced with the question of how to build smarter machines for a variety of tasks, such as computer vision, speech recognition, or machine translation. State-of-the-art systems are often comprised of a complex set of interdependent components. For instance, the computer vision task of localizing people in images [1] involves detecting parts of a person, spatial reasoning among these parts, and contextual modeling. These complex systems have led to significant progress in the field. However, the relative importance of each component is debatable. Hence it remains unclear what steps should be taken to make further progress. Which component should we focus on? Do we need a new pipeline all together?

To answer these questions, we introduced a paradigm called "human-debugging". We leverage numerous human studies to systematically debug computational models and identify performance bottlenecks. For each specific component in the computational system's pipeline, we replace the machine's output with output generated by human subjects. To ensure fair comparisons, it is important that the information used as input by the human subjects is the same information given to the machine. The relative performance of the complete system is then compared using various combinations of machine and human components to find those critical to accuracy. For example, in a person detection system, instead of using a state-of-the-art machine classifier to detect parts, we used human subjects to act as part detectors by labeling small image patches as a person's head, foot, etc. Crowdsourcing makes such large-scale human studies possible. In addition, by using human subjects to independently solve each component, we can estimate the potential for improvement a specific pipeline holds. Human-debugging helps focus future research endeavors in directions that really *matter*.

Humans are a working system whose performance we wish to replicate. It seems rather strange that most systems utilize humans only for providing training data with coarse labels. Human-debugging allows for a deeper involvement of humans in advancing AI. In the following section, we describe two instances of this paradigm and the resultant findings. In Section 3 we describe some challenges we face while employing this paradigm.

2 Human-Debugging

Our proposed human-debugging paradigm is applicable to any field in AI such as computer vision, machine translation, speech recognition, natural language processing, etc. For speech recognition, one could evaluate the relative importance of acoustic and language models. A similar analysis



Categorize an image

Shown below are example patterns from each of the eight categories.

Category 1 Category 2 Category 3 Category 4

Category 5 Category 6 Category 7 Category 8

For the pattern shown below, please select the category that it belongs to.

○ Category 1 ○ Category 2 ○ Category 3 ○ Category 4
○ Category 5 ○ Category 6 ○ Category 7 ○ Category 8

Submit

(a) Human studies interface

(b) Influence of different factors

Figure 1: (a) Snapshot of our human studies interface used in [2]. (b) The F-ratio computed using analysis of variance (ANOVA) on several datasets. We can see that in almost all cases, the features have the highest influence on the performance.

would be applicable to machine translation. We have applied this paradigm to several computer vision tasks [2–6]. We have evaluated the role of different factors at various levels of granularity, ranging from high-level factors such as representation, learning algorithms and amounts of training data [2] to specific components such as part detection and spatial reasoning in a particular person-detection pipeline [3].

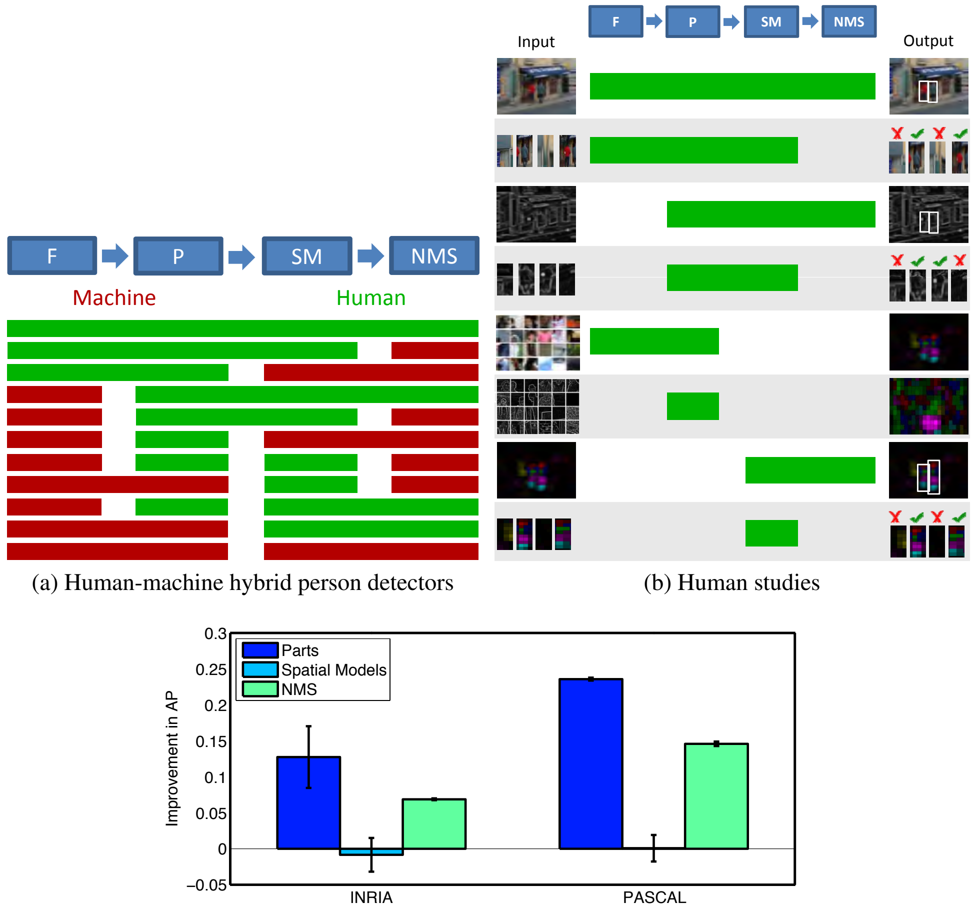
2.1 Features, Algorithms, or Data?

Our first task evaluated the role of image representations, learning algorithms and amount of training data for image classification [2]. We performed a variety of human studies and machine experiments to examine whether human learning and pattern matching algorithms are significantly better than some of today's popular classification strategies. The amount of training data and feature types are varied, and their affects on accuracies studied. In human studies, to prevent the use of prior knowledge about images by the subjects, we do not display to them any direct image information such as texture patches or color. Instead, we use abstracted visual patterns as stimuli as seen in Figure 1 (a). Such an abstract pattern can be generated for any feature vector. We presented identical learning tasks *i.e.* the same feature vectors for training and testing, to machines and humans. Having human subjects solve precisely the problems posed to machines allows us to draw meaningful conclusions. We experimented with a variety of standard image classification datasets, image representations, classifiers, and amount of noise in the data. Our human studies were performed on Amazon Mechanical Turk.

In our experiments we find no evidence that the human learning algorithm is better than the standard machine learning algorithms popular today. Moreover we find that humans do not benefit much from more training data. As a result, we hypothesize feature representation as the factor that gives humans an advantage over machines. In fact, through multi-way analysis of variance (ANOVA) (see Figure 1 (b)), we find that the choice of feature representations impacts recognition accuracy the most as compared to the other factors. Focussing on this factor more, we have also studied the roles of local and global information in image representations [4], and the roles of appearance and contextual information [5] for image labeling.

2.2 Features, Parts, Spatial Models, or Context?

Our second task explores the problem of detecting people in images. A state-of-the-art parts-based person detector [1] can be roughly broken into four components: feature detection, part detection, spatial part scoring and contextual reasoning including non-maximal suppression. We wish to determine which component if improved may lead to the largest boost in overall system accuracy. Our experiments [3] assembled numerous combinations of components performed by humans and



(a) Human-machine hybrid person detectors

(b) Human studies

(c) Results

Figure 2: (a) We replaced each component in the machine pipeline (red) with human subjects (green). (b) The various tasks performed by human subjects. For instance, in the first task (top) subjects performed the entire person detection process by looking at an input image, and providing bounding boxes around people in the image. In the remaining tasks, subjects only perform a part of the process as denoted by the extent of the green bars. (c) Summary of our findings.

machines to form complete object detectors, as seen in Figure 2 (a). The various human tasks involved are summarized in Figure 2 (b). As before, we conducted these human studies on Amazon Mechanical Turk.

Our experiments concluded that part detection is the weakest link for challenging person detection datasets. Non-maximal suppression and context can also significantly boost performance. However, the use of human or machine spatial models does not significantly or consistently affect detection accuracy. A summary of the results can be seen in Figure 2 (c). This was the first analysis of its kind that provided the community valuable and concrete feedback about which specific problem to focus on to improve overall performance: in this case, classifying local image patches into one of six person-part categories.

3 Challenges

The key idea behind human-debugging is to replace isolated components of a machine pipe-line with human subjects. This necessitates designing studies that require humans to perform very specific tasks; whose input and outputs precisely match those used by the equivalent machine implementation. This leads to several interesting challenges.

Accessing isolated human-models: It is crucial for the information available to humans to be equivalent to that available to the machine implementation. This often involves providing information in

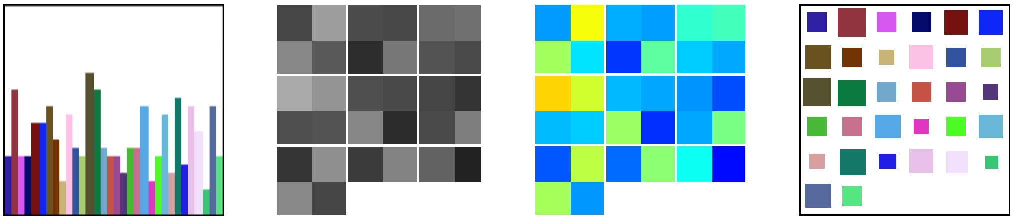

(a) Colored-bars (b) Intensity-map (c) Heat-map (d) Colored-squares

Figure 3: Example visualizations of a 32 dimensional feature vector. The value of each of the 32 entries in the feature vector is converted to the height of the 32 bars in (a), the intensity of each of the 32 blocks in (b) and (c), and the area of each of the 32 squares in (d).

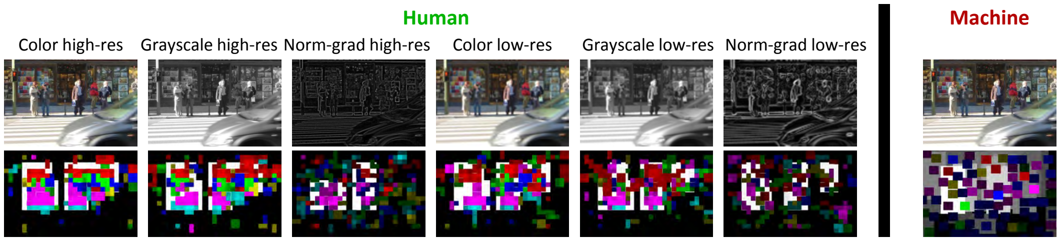


Figure 4: Part-detection visualizations created for human and machine detected parts.

a manner that does not allow humans to use their prior knowledge about the world. For instance, for one of the experiments in [2], we wished to show subjects only the texture information in an image as extracted by a machine. If we simply show subjects gray-scale images without color, they not only have access to the texture information but also very high-level information such as objects present in the image, etc. How can we show humans *only* the texture information in an image? This necessitates the use of abstract visualizations.

Note that this challenge is more exaggerated in some AI domains like vision than in others. In speech recognition or machine translation, one can find human subjects that do not know a certain language and are thus devoid of high-level knowledge specific to that language. This is not quite feasible in vision, where most human subjects share the same visual world priors.

Visualizing high-dimensional data: The input to machines is often high-dimensional data (*e.g.* texture histogram of an image). How do we create an abstract visualization of this data to show human subjects? In [2], we experimented with several visualizations as shown in Figure 3. We found that for a particular task, subjects performed at 34% using 'Colored-bars', 47% using 'Intensity-map', 50% using 'Heat-map and 47% using 'Colored-squares'. It is difficult to design a high dimensional visualization that does not introduce its own biases. For instance, the 'Heat-map' visualization biases subjects to believe that features that are nearby in 2D are correlated.

In [3], in order to display the person part-detections to human subjects, we created a visualization where each patch in the image was colored with one of six-colors corresponding to the six parts we considered: head, torso, arm, hand, leg, foot. The intensity of the color corresponds to the confidence in the classification of the patch. Human subjects were trained on these visualizations, and the effectiveness of their spatial models was evaluated. Example visualizations are shown in Figure 4. It is an open question whether a better visualization would lead to improved results for the human subjects.

Invoking natural visual pathway: Unfortunately, when working with such abstract visualizations we can not ensure that the tasks posed to humans are natural. The resultant performance is clearly a function of the chosen visualization. Hence, an ideal visualization would be one that invokes the natural visual pathway in human subjects resulting in optimal human performance at the specified task; while still allowing explicit control over what information is made available to subjects. Chernoff [7] suggested the use of faces to display k-dimensional points. Humans are known to be sensitive to variations in facial features. However, this is applicable only for $k \leq 18$ feature vectors. Are there other natural visualizations that can encode higher-dimensional data? Most existing visualizations for high-dimensional data are not 'natural' [8].

References

[1] P. F. Felzenszwalb, R. B. Girshick, D. McAllester and D. Ramanan. Object Detection with Discriminatively Trained Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2010.

[2] D. Parikh and C. L. Zitnick. The Role of Features, Algorithms and Data in Visual Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[3] D. Parikh and C. L. Zitnick. Finding the Weakest Link in Person Detectors. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[4] D. Parikh. Recognizing Jumbled Images: The Role of Local and Global Information in Image Classification. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[5] D. Parikh, C. Zitnick and T. Chen. From Appearance to Context-Based Recognition: Dense Labeling in Small Images. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[6] C. Li, D. Parikh and T. Chen. Extracting Adaptive Contextual Cues from Unlabeled Regions *International Conference on Computer Vision (ICCV)*, 2011

[7] H. Chernoff. The Use of Faces to Represent Points in K-Dimensional Space Graphically. *Journal of the American Statistical Association*, 1973.

[8] A. Hinneburg, D. A. Keim and M. Wawryniuk. HD-Eye: Visual Mining of High-Dimensional Data. *IEEE Computer Graphics and Applications*, 1999.