Science Robotics
AAAS

robotics.sciencemag.org/cgi/content/full/4/37/eaay4663/DC1

Supplementary Materials for

A tale of two explanations: Enhancing human trust by explaining robot behavior

Mark Edmonds*, Feng Gao*, Hangxin Liu*, Xu Xie*, Siyuan Qi, Brandon Rothrock, Yixin Zhu*,
Ying Nian Wu, Hongjing Lu, Song-Chun Zhu*

*Corresponding author. Email: markedmonds@ucla.edu (M.E.); yixin.zhu@ucla.edu (Y.Z.); sczhu@stat.ucla.edu (S.-C.Z.)

Published 18 December 2019, *Sci. Robot.* **4**, eaay4663 (2019)
DOI: 10.1126/scirobotics.aay4663

The PDF file includes:

Other Supplementary Material for this manuscript includes the following:

(available at robotics.sciencemag.org/cgi/content/full/4/37/eaay4663/DC1)

Movie S1 (.mp4 format). Example explanation video of full model shown to human participants.
Movie S2 (.mp4 format). Example haptic explanation video shown to human participants.
Movie S3 (.mp4 format). Example symbolic explanation video shown to human participants.
Movie S4 (.mp4 format). Example text explanation video shown to human participants.
Movie S5 (.mp4 format). Example baseline video shown to human participants.
Data S1. Final data files (.zip).
Code S1. Software files (.zip).

Supplementary Materials

Text S1. Additional model results

Figure S1 presents additional generalization experiment results using the improved GEP. Each augmented 3D-printed cap generates a significantly different time-series haptic signal, indicating the haptic interactions are substantially different from one another. These results demonstrate the GEP's ability to transfer to bottles with haptic signals that are different from the ones in the demonstration.

We qualitatively analyze one example in Figure S2 to further justify that the generalization scenarios are significantly different from the testing scenarios by comparing the haptic signals between the two. Specifically, given one haptic signal in testing and one in generalization performing the very same action, we treat these two sets of haptic signals as time series data and estimate the trend using kernel methods. After obtaining the trend, a rigorous testing procedure (*33*) for evaluating whether these two haptic signals have the same distribution is performed by comparing the L2 distance between the curves. The results indicate that the haptic signals of all bottles are significantly different from one another; Δ is not closed to 0. A comprehensive, quantitative analysis of more haptic signal data is presented as a confusion matrix in Figure S3).

We also note that an alternative model commonly used for this type of analysis is the ARIMA method. However, our observations (haptic signals) are mean non-stationary, which is not suitable for ARIMA; ARIMA works well for an integrated process.

Text S2. Additional materials and methods

Text S2.1. Model limitations

Generating a smooth action sequence mapped from a human demonstration to a robot is a non-trivial task. In this paper, for the symbolic planner, we simplified this process by assuming each mapped action is executable by the Baxter robot; each atomic action or motion primitive (terminal node in the grammar) is designed, not automatically learned. However, this should not impact the overall contributions presented in the paper, as the mapping in a supervised fashion does not affect the experiments for evaluating human trust. In addition, this embodiment problem is solvable in some instances if we introduce the concept of "mirroring" (*34*).

Our approach assumes that each robot action corresponds to an equivalent human action. However, if adopted after learning the grammar, trajectory optimization methods (*e.g.*, CHOMP (*35*), STOMP (*36*), and TrajOpt (*37*)) could improve the action/behavior of the robot to generate smoother action sequences, or even produce different actions that are not the same as the human demonstrations. In addition, a grammar can, in fact, generate sequences that are not seen in demonstrations because of the compositional nature of the grammar rules; in other words, it is possible for the robot to solve the tasks using different action sequences from human sampled from the grammar model. Nevertheless, a grammar has no inherent mechanism for the robot to discover entirely new actions for the task.

Text S2.2. Training details of embodied haptic model

In this section, we present the implementation detail for reproducibility.

Network Architecture The autoencoder is constructed with a multi-layer perceptron (MLP); see Table S2. The human embedding can be obtained with a forward pass through the network. The supervision for the autoencoder is the original human post-condition. The loss is measured

by the reconstruction error. The robot-human embodiment mapping is implemented with an MLP; see Table S3. The embodiment mapping is trained using equivalent human and robot post-conditions (equivalent here means the post-condition of executing the same action successfully). The human post-condition is fed through the autoencoder to produce a human embedding, and this embedding serves as the supervision target for the embodiment mapping network. The last major component of the embodied haptic prediction model is the action predictor, also implemented with an MLP; see Table S4. The supervision for the action predictor is the ground-truth human action labels.

Training Details We adopt a two-step updating schema for the embodied haptic model. In the first step, we feed forward the human post-condition data into the autoencoder. The encoder will reduce the high-dimensional human data to a low-dimensional human embedding; the encoder and the decoder are learned with hyper-parameter shown in Table S5. The supervision for the autoencoder is the reconstructed original human post-condition. In the second step, with the human embedding and the action labels, the action predictor and the embodiment mapping are training jointly with the hyper-parameters shown in Table S5. The embodiment mapping is trained using equivalent human and robot post-conditions (equivalent here means the post-condition of executing the same action successfully). The human post-condition is fed through the autoencoder to produce a human embedding, and this embedding serves as the supervision target for the embodiment mapping network. The supervision for the action predictor is the ground-truth human action labels.

Text S2.3. Details on tactile glove

In this paper, we use 15 IMUs to obtain the relative poses of finger phalanges with respect to the wrist (see Figure S6A) and develop a customized force sensor using a soft piezoresistive mate-

rial (Velostat) whose resistance changes under pressure (*28*). The 26 force sensors are placed on the palm and fingers, as shown in Figure S6B. The force sensor is constructed in a 5-layer, mirrored structure—Velostat is the inner layer, conductive fabric and wires are the middle layers, and insulated fabric is the outer layer. Figure S6C illustrates the structure of the force sensor, and the force-resistance relation is characterized as Figure S6D. In total, the glove provides 71 degrees of freedom, including all pose and force measurements of the hand, resulting in a fine-grained reconstruction. The relative poses between the hand and manipulating objects (bottles and caps) are captured by a Vicon motion capture system. We capture 64 demonstrations in total; the number of demonstrations varies by the number of possible grasping approaches human demonstrators found natural. Twenty-nine demonstrations were collected for the Bottle 1, 23 for Bottle 2, and 12 for Bottle 3.

Text S2.4. Force visualization

To visualize the forces imposed by the robot gripper, we first identify the max force magnitude in all the force signal data collected from human demonstrations. Then, all force data is normalized to the value between 0 and 1, where 0 corresponds to pure green in the visualization, and 1 pure red. The value in between is interpolated linearly and displayed on the robot's palm.

Text S2.5. Additional human experiment details

The prediction phase evaluates how well each explanation panel imparts prediction ability after observing a robot's behaviors in solving the problem of opening a medicine bottle. Note that during the familiarization phase, the robot explains its behavior through explanatory panels, but during the prediction phase, subjects observe the robot executing the task with only the RGB videos. Thus our prediction question asks "*after familiarizing with explanatory panels, how well are human subjects able to predict robot behavior when observing only RGB robot exe-*

cutions?" The prediction accuracy is computed as the percentage of correct action predictions in the sequence. This experimental design examines how well each explanatory panel imparts prediction ability under new robot executions where no explanation panel is available. For each question, participants selected from 8 actions: *push on the cap*, *pinch the cap*, *pull the cap*, *twist the cap*, *grasp the cap*, *ungrasp the cap*, *move the left robot arm to grasping position*, and *nothing*.

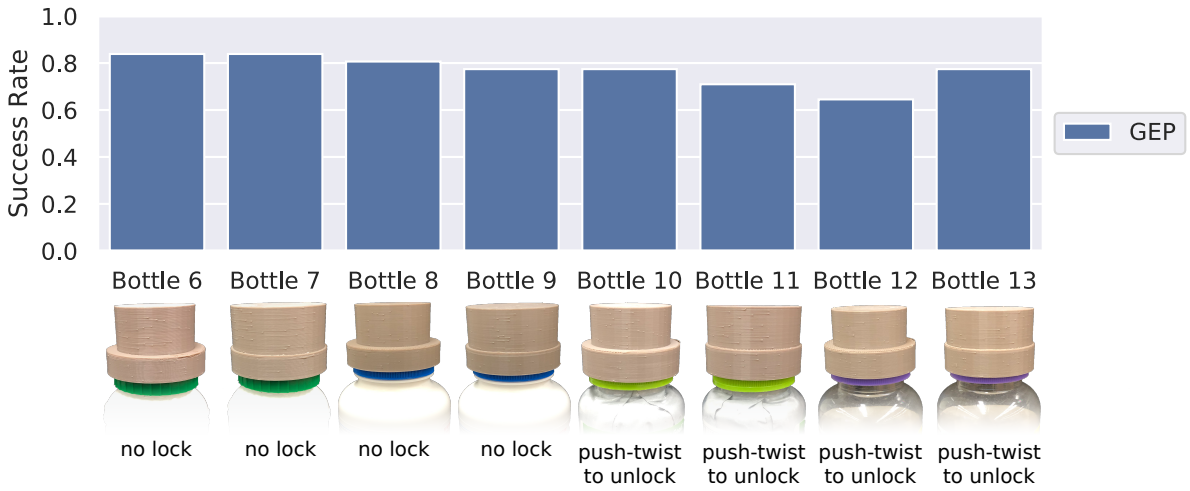

Fig. S1. Additional generalization experiments on bottles augmented with different 3D-printed caps. The GEP shows good performance across all bottles, indicating the GEP is able to generalize to bottles with similar locking mechanisms as in the human demonstrations, but significantly different haptic signals.

Table S1. Numerical results and SDs for human participant study; the same data was used in Figure 7.

	Qualitative Trust		**Prediction Accuracy**	
	Mean	Std. dev.	Mean	Std. dev.
Baseline	71.7	16.8	0.481	0.176
GEP	82.6	17.5	0.644	0.202
Symbolic	81.9	17.4	0.641	0.228
Haptic	75.7	16.2	0.541	0.231
Text	70.1	22.7	0.593	0.218

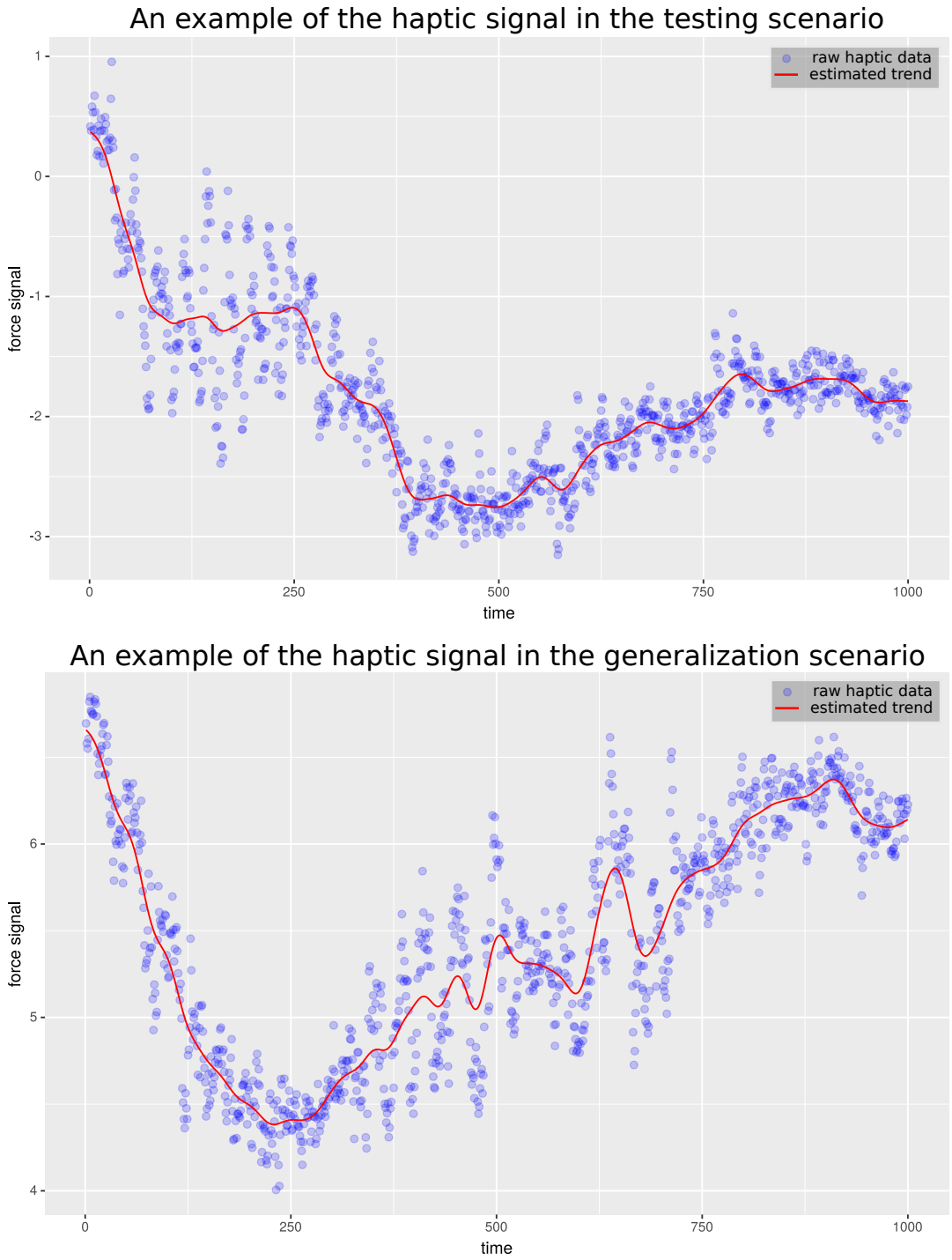

Fig. S2. Examples of estimated trends for testing and generalization haptic data. Examples of aligned haptic signals in time used in testing (top) and generalization (down) data. The haptic data was collected by executing the same actions on various bottles in testing and generalization scenarios. Light blue dots denotes raw noisy haptic signals. The solid red line denotes the estimated trend for statistical analysis.

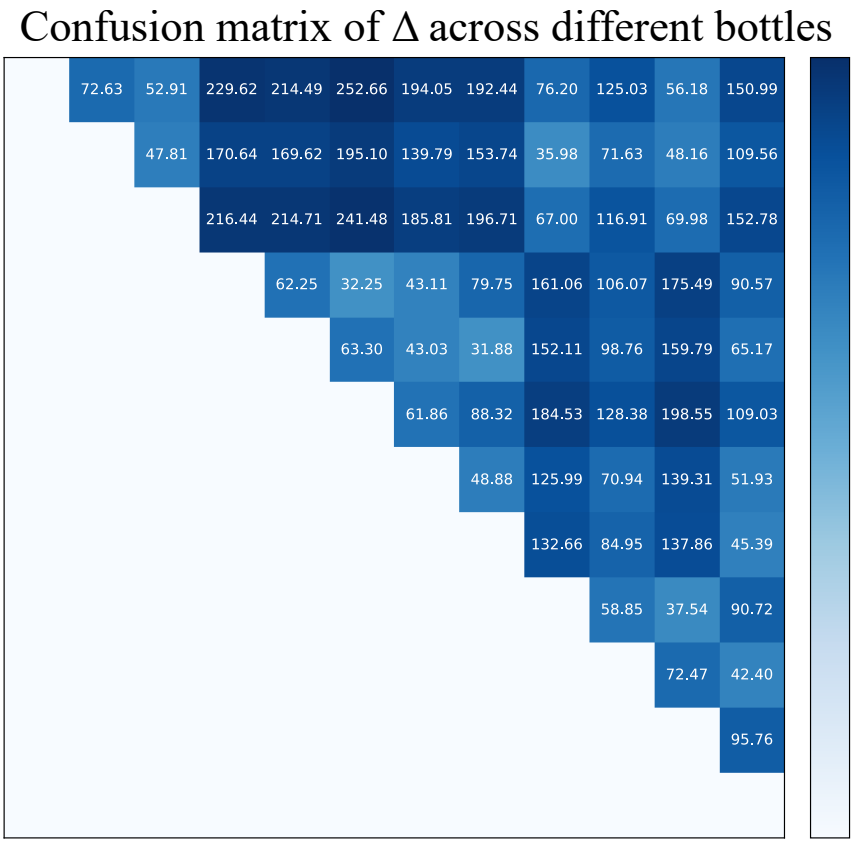

Fig. S3. The confusion matrix of Δ across different bottles based on the haptic signals. Higher Δ values indicate lower similarity.

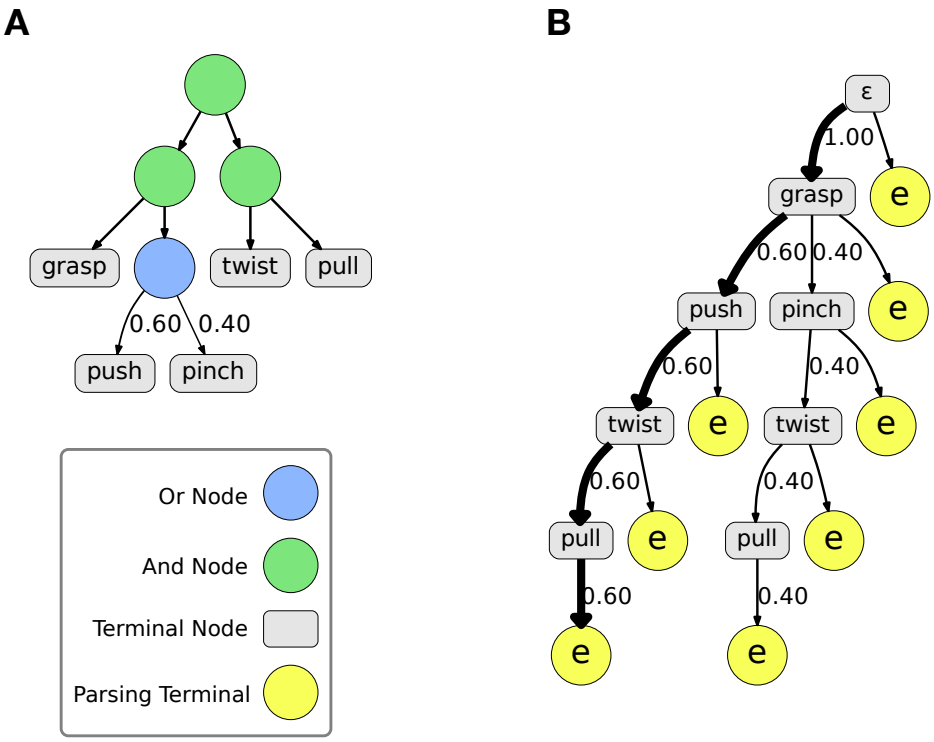

Fig. S4. An example of action grammars and grammar prefix trees used for parsing. (**A**) An example action grammar. (**B**) A grammar prefix tree with grammar priors. The numbers along edges are the prefix or parsing probabilities of the action sequence represented by the path from the root node to the node pointed by the edge. When the corresponding child node of an edge is an action terminal, the number along the edge represents a prefix probability; when the corresponding child is a parsing terminal e, the number represents the parsing probability of the entire sentence. In this example, the action sequence "grasp, push, twist, pull" has the highest probability of 0.6. The root ϵ represents the empty symbol where no terminals were parsed.

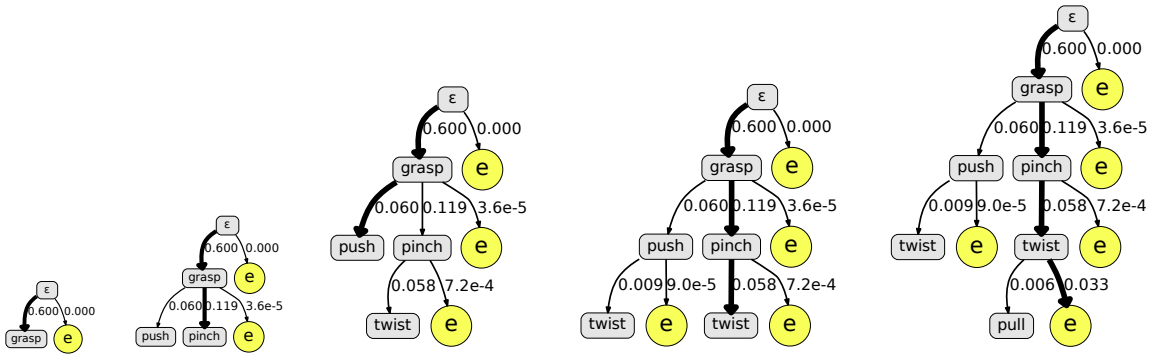

Fig. S5. An example of the GEP. An illustration of the parsing process of the Generalized Earley Parser (GEP). It performs a heuristic search in the prefix tree according to the prefix/parsing probability. It iteratively expands the tree and computes the probabilities as it expands the tree. The search ends when it hits a parsing terminal *e*. The paths in bold indicate the best candidates at each search step.

Algorithm S1. Algorithm of the improved GEP for robot planning.

Input : Grammar G, Haptic Model H, Maximum Step T
InputStream : Haptic Signal f_t
OutputStream: Robot Executable Action a_{t+1}

$t = 0$
Initialize empty matrix y_0
while $t <= T$ **do**
  if $t == 0$ **then**
    $p(a_{t+1}) =$ uniformVector()
  else
    $f_{t+1} =$ getHapticSignal()
    $p(a_{t+1}) =$ hapticPlanner$(f_t, a_t; H)$ // Equation 1
  end
  $y'_{t+1} = [y_t; p(a_{t+1})]$ // Append probability vector to one-hot matrix y_t
  $a_{t+1} =$ prefixSearch(G, y'_{t+1}) // Equation 7
  $y_{t+1} = [y_t;$ oneHot$(a_{t+1})]$ // Extend one-hot matrix from y_t to y_{t+1}
  executeRobotAction(a_{t+1})
  if *goalAchieved()* **then break**
  $t = t + 1$
end

Table S2. Network architecture and parameters of the autoencoder. Network architecture is defined from the top of the table to the bottom, with the first and last layer being input and output, respectively.

Operator	Params
Linear	80
ReLU	
Linear	64
ReLU	
Linear	16
ReLU	
Linear	8
ReLU	
Linear	16
ReLU	
Linear	64
ReLU	
Linear	80

Table S3. Network architecture and parameters for robot to human embedding. Network architecture is defined from the top of the table to the bottom, with the first and last layer being input and output, respectively.

Operator	Params
Linear, Linear	3, 1
ReLU, ReLU	
Linear, Linear	128, 128
ReLU	
Linear	8

Table S4. Network architecture and parameters for action prediction. Network architecture is defined from the top of the table to the bottom, with the first and last layer being input and output, respectively.

Operator	Params
Linear, Linear	8, 13
ReLU, ReLU	
Linear, Linear	64, 64
ReLU	
Linear	10

Table S5. Hyperparameters used during training.

Parameter	Value
Autoencoder learning rate	5e-5
Action predictor learning rate	5e-5
Balance param. (β)	1
Batch size	16
No. of epochs	150

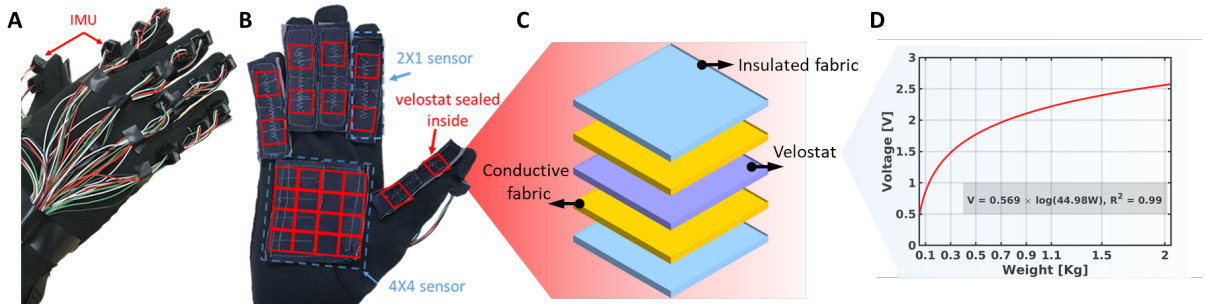

Fig. S6. Tactile glove hardware design. (**A**) The dorsum of the tactile glove developed for the present study consisting of 15 IMUs. (**B**) 26 integrated Velostat force sensor on the palmar aspect of the hand. (**C**)The structure of the force sensor. (**D**) Characteristics of the forcevoltage relation, which is described by a logarithmic law of the force sensor.

Table S6. Specifications of the computing platform used in the experiments.

Parts	Description
Robot	Baxter
Manipulator	Right: ReFlex TackkTile gripper. Left: Robotiq S85 parallel gripper
Computer	ZOTAC ZBOX-EN1070K: i5-7500T with GTX 1070
Vision sensor	Kinect v2

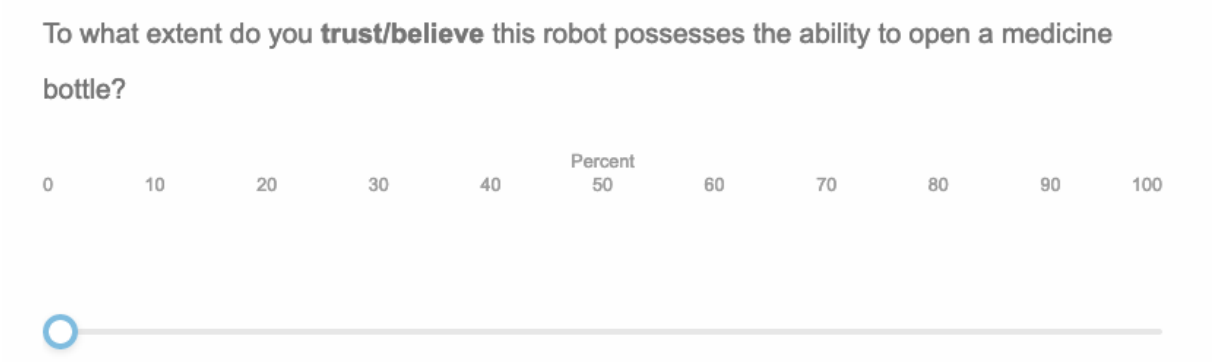

Fig. S7. Qualitative trust question asked to human participants after observing two demonstrations of robot execution. This question was immediately asked after the familiarization phase of the experiment; in other words, we asked this question immediately after the subjects had observed robot executions *with* access to the explanation panel (if the subject's group had access to an explanation panel; *i.e.* all groups except baseline).

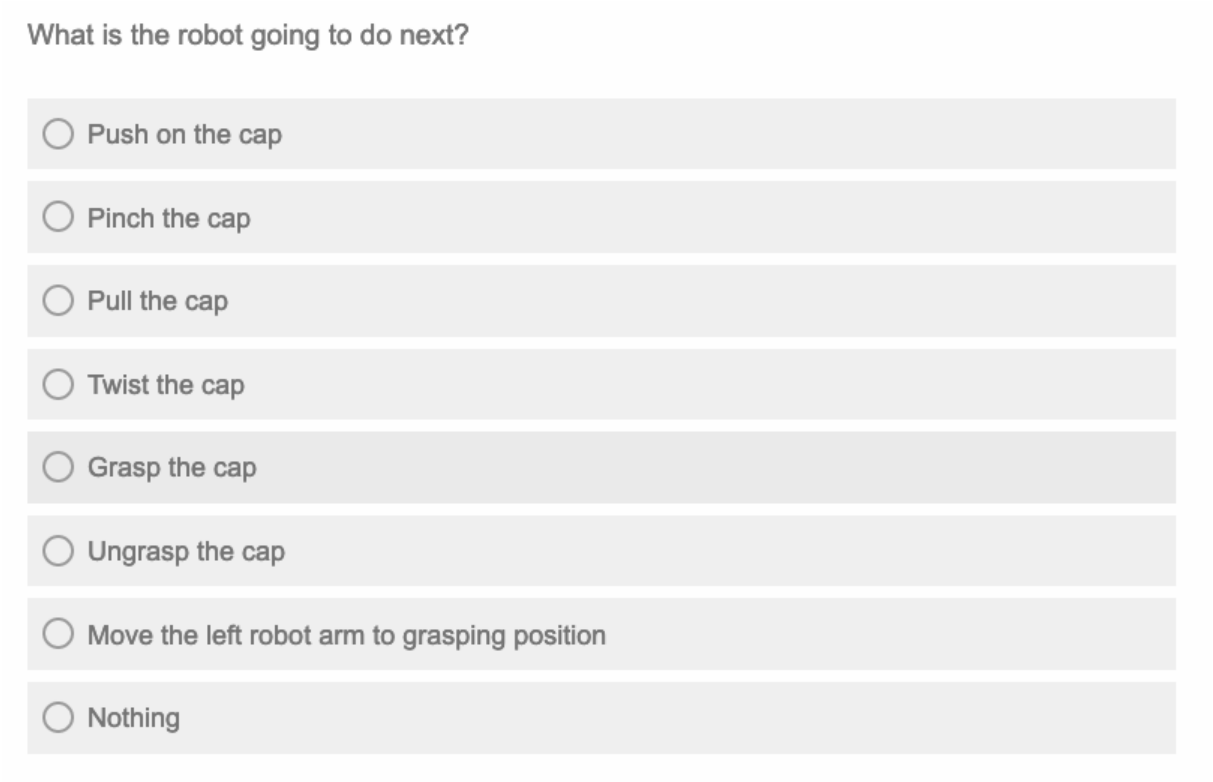

Fig. S8. Prediction accuracy question asked to human participants after each segment of the robot's action sequence during the prediction phase of the experiment. No group had access to explanation panels during the prediction phase; subjects had to predict the action while only observing RBG videos of each action segment.