Proceedings of the Eighth AAAI Conference on Human Computation and Crowdsourcing (HCOMP-20)

Predicting Crowdworkers' Performance as Human-Sensors for Robot Navigation

Nir Machlev, David Sarne
Bar-Ilan University, Israel
nirmachlev@gmail.com, sarned@cs.biu.ac.il

Abstract

This paper provides and evaluates a new paradigm for collaborative human-robot operation in search and rescue-like settings with information asymmetry. In particular, we focus on settings where the human, a crowdworker in our case, is used as a sensor, providing the route-planning module with essential environmental information. In such settings, the ability to predict the expected performance of the collaborating crowdworker in real-time is instrumental for maintaining a continuously high level of performance. Through an extensive set of experiments with crowdworkers recruited and interacted through Amazon Mechanical Turk, we show that effective online prediction is indeed possible, however only if distinguishing between two subpopulations of crowdworkers, termed ”operators” and ”sensors”, applying a different prediction model to each. Furthermore, we show that even the classification of crowdworkers to the two types can be carried out successfully in real-time, based merely on the first two minutes of collaboration. Finally, we demonstrate how the above abilities can be used for a more effective workers' recruiting process, resulting in a substantially improved overall performance.

Introduction

Autonomous robots are on the rise nowadays and the list of domains and applications where they are used has become quite impressive. Examples are numerous, and include, among others, surveying the environment for threats (Lösch et al. 2018), cleaning up the pollution in oceans (Rahmawati et al. 2019), mapping (Faessler et al. 2015), vacuuming and cleaning (Jones et al. 2019) and playing soccer to entertain audiences (Kitano et al. 1997). Common to all these robots, which vary significantly in size, functionality, mobility and cost, that they can be programmed to perform some tasks with little to no human intervention or interaction. Still, when the environment they are operating in is highly dynamic and unpredictable, it may become too complex for them to decipher and understand the happenings (Winter et al. 2017). For example, consider indoor localization and navigation, where seemingly the environment is highly structured. Even here, various unexpected can happen, e.g., electronic doors that fail to open as they should, obstacles completely blocking the path, and counter-intuitive (room or level) numbering systems such as a missing 13th floor in a building (Tomko and Richter 2015). The help of a person can thus take either the form of resolving problems, help in decision making regarding the preferred course of operation, and providing environmental information that cannot otherwise be collected or properly interpreted by the robot (Ghandour et al. 2016).

In this paper we consider the setting where the robot depends on a person to provide it with temporal task-related crucial information unfolding along with the task, that is otherwise unavailable to it. In particular, we focus on domains where the nature of the task is robot-navigation, yet route planning heavily relies on knowing what events take place in the space (or plain) along time. For example consider robotic exploration missions (Sampedro et al. 2019; Wang et al. 2011) such as Urban Search and Rescue (USAR) (Nourbakhsh et al. 2005) or planetary exploration (Apostolopoulos et al. 2001). Here, the interpretation of images taken by the robot's sensors typically requires the visual perception skills of humans (Kosti, Sarne, and Kaminka 2014), e.g., identify the remainings of a crashed airplane in the clutter, point to suspected locations where an intruder may be hiding and identify survivors in a disaster area. Still, once this information is transferred to the robot, the mission planning would be better carried out by the robot as its processing capabilities are superior. Meaning that the collaborating person functions as a human sensor rather than a robot operator (Lewis et al. 2009).

Whenever the task of acting as a human sensor does not require much expertise and skill, it can be easily outsourced to crowdworkers in crowdworking platforms (e.g., Amazon Mechanical Turk), benefiting from their high availability and relatively cheap cost (compared to a trained operator). Moreover, crowdwokers can be employed on an ad-hoc basis, and easily replaced on-the-fly if deciding that they are not efficient enough. Alongside the many advantages of using crowdworkers as human sensors, some disadvantages cannot be ignored. Workers may vary substantially in their capabilities and pace. They may fail to understand the task. They are typically less committed to the task, compared to permanent workers, and at times attempt to increase their revenues by working on several tasks in parallel, paying less attention to each (Elmalech et al. 2016). Furthermore, performing as a sensor is a rather monotonous task and people have an inherent difficulty to remain focused in such (Krueger 1989). Therefore having the ability to identify poor performance on the fly or predict if a worker is likely to perform poorly throughout the task is crucial for the task's success, as poorly performing working can be replaced.

In this paper, we report the results of attempts to develop models for predicting crowdworkers' performance as a sensor aiming to enhance a robot's route planning. This kind of prediction is highly challenging, as it needs to be carried out based on a short interaction with the worker, and even for that short interval we do not have any measure reflecting the worker's efficiency.[1] For example in USAR applications, the system may realize in real-time whether in locations visited based on an indication received from the worker there were indeed survivors but cannot tell if there were any in locations not indicated as such. Therefore, even if one's performance in a short interval of operation is fully indicative for her performance throughout the task (which is certainly not obvious, since, as mentioned above, performance may deteriorate over time due to boredom or shifts in the worker's attentional state), reasoning about performance in that short interval is not trivial by itself (Nimni and Sarne 2020).

As an experimental infrastructure, we use a system emulating the use of a robotic boat aiming to deter pelicans from fish ponds, a problem affecting farmers worldwide. Here, the goal of the human sensor is to identify pelicans as they arrive and provide their location information to the boat. For the prediction models, we use a rich set of features that can be collected in the above-mentioned environment and four quite standard machine learning methods: Linear Regression, Random Forest, AdaBoost, and SGD. Alas, we find that the resulting prediction is poor and quite noisy. Interestingly, we find that effective prediction can be executed by distinguishing between two subpopulations of human sensors and carrying out the prediction for each of the two separately. Workers of the first population indeed function as instructed and provide a proper indication of events based on their best effort. Meaning that the differences in their performance are fully attributed to how attentive they are to the task and how hard they try. Workers of the second population, on the other hand, limit the number of events they report at any given time, attempting to take full control over the robot's route rather than benefiting from the robot's route-planning capabilities. The prediction models developed are complemented by an effective classification model that enables classifying a new worker to one of the two groups merely based on the first two minutes of operation. This allows us to decide in real-time if we want to continue employing this worker or replace her with a newly recruited one. Applying such a dynamic allocation mechanism on workers in our experiments results in 67 percent performance improvement, compared to the use of static assignment.

Related Work

The research reported in this paper is within the intersection between human-robot interaction (HRI) and crowdsourcing. HRI research suggests a plethora of designs aiming to improve an operator's control of a robot (Sheridan 2016). One aspect of the interaction that is especially important in our case is autonomy, in particular the level the human and the robot interact and the degree to which each is capable of acting autonomously, on a scale ranging from direct control to peer-to-peer collaboration. while with direct control the focus is on reducing the cognitive load of the operator, in the fully collaborative scheme it is on creating robots with the appropriate cognitive skills to interact naturally or efficiently with the operator (Goodrich, Schultz, and others 2008), the way information is exchanged between the human and the robot, and the amount of situation awareness produced by the interaction (Endsley 2016). Within the above line of work, the closest to our model is the work of Bruemmer et al (2004), focusing on evaluating mixed-initiative control methods for novice operators of a search and rescue robot. One particular finding of their work is that the system usability can be enhanced through the addition of navigational autonomy, freeing the user to focus on the search and rescue task instead of robot navigation. This is exactly the mode of operation we use in the research reported in this paper.

Many designs for robot-control incorporated the principle of adjustable autonomy, which enables a system to operate in different autonomic conditions and transfer control to or between the system's operators (Mostafa, Ahmad, and Mustapha 2019). The transfer of control can be beneficial whenever the robot is unable to perform the task with complete autonomy (Zilberstein 2015) or when the importance of the task makes human intervention crucial (Pollack, Tsamardinos, and Horty 1999). In particular, a robot can benefit from the user's experience and skill, having her guide decision making whenever full enumeration of subspaces of the full problem space is impractical, e.g., in planning and scheduling (Alexander, Raja, and Musliner 2008). Similar ideas can be found in Human-centered automation (HCA), which offers the benefit of partially automating a system when full automation is not possible, leaving the difficult-to-automate parts of the system to humans hence increasing operator acceptance of an autonomous system (Dorais and Kortenkamp 2000). The main question however, in all the above line of work, is when to transfer decision-making control from the robot to the user (Scerri, Pynadath, and Tambe 2002) and to what extent.

With the vast interest in crowdsourcing platforms, the use of crowdworkers for enhancing robot operation is gradually increasing. For example, Moradi et al (2016) suggest the use of crowdworkers' human intelligence to train soccer robots and improve their decision making process. Almosalami et al (2018) use crowdsourcing-based interface in order to ask people to operate a garbage-collecting robot at beaches. Zhao et al (Zhao et al. 2019) use crowdsourcing to help a robotic hand synthesize human physical skills. Much like in our case, many of these applications use crowdworkers in order to provide the system with otherwise unavailable information. For example, Diamantas (2020) presents a new

[1] If having the ability to properly identify the events guiding the route planning, there would not be a need for the human's help in the first place.

approach for resolving the kidnapped robot problem by asking followers on Twitter for information about its location, and Chung et al (Chung et al. 2019) use crowdworkers to alert the system if a simulated autonomous car is about to make an accident.

Finally prior work provides much evidence for the fact that crowdworkers are likely to employ a wide variety of interaction styles with robots (Breazeal et al. 2013). However alongside modes of control, there are various other considerations that were found to be instrumental in using crowdworkers in general, that should be taken into consideration. For example,

while crowdsourcing platforms enable collecting fast and cheap data, its quality is often questionable (Mason and Suri 2012). Furthermore, crowdworkers are human, and long sequences of the same monotonous tasks might intuitively reduce the quality of their work (Krueger 1989). Several approaches have been suggested in recent years for increasing crowdworkers attention to the task, for example the use of diversions containing small amounts of entertainment (Dai et al. 2015) and the generation of dummy (artificial) events throughout the task (Elmalech et al. 2016).

Model

We consider a standard search and rescue-like setting where a robot needs to reach and handle events of a spatial nature (Ito and Maruyama 2016; Lewis, Sycara, and Nourbakhsh 2019). New events occur on a timely basis and are characterized by their location in the plane and the amount of time they remain active. To handle an event, the robot needs to physically reach its location while it is still active. The goal of the robot is to handle as many events as possible, overall.

The robot is fully capable of navigating in the plane. Furthermore, it is equipped with a planning module that updates its route in real-time as new (full or partial, accurate or noisy) information about current active events in its environment unfolds. Still, despite its many capabilities, it is unaware of the appearance of events and their locations. In order to receive such information the robot has to rely on a human collaborator, who functions as a human sensor, recognizing events on the fly and conveying this information to the robot through a designated interface. The event-recognition task is quite intuitive and can be performed by crowdworkers, enabling various advantages associated with such ad-hoc employment as discussed above. In particular, the robot can easily recruit a new crowdworker, replacing an existing one in case it believes the latter performs poorly as a human sensor. The goal of the research reported in this paper is thus to develop effective prediction models for crowdworkers' performance as a human-sensor, supporting such crowdworker's replacement on the fly.

Experimental Framework

As a testbed for our experiments, we used an experimental framework simulating GPS-based robotic boat navigation in fish ponds. The framework was developed as part of a larger multi-institute collaboration aiming to provide solutions for deterring massive fish-eater birds from the depredation of fish ponds. Deterrence of migrating birds from fish ponds is a fundamental challenge for ecologists, ornithologists, birdwatchers and fish farmers across the world. These birds will eat either the fish or the fish food, resulting in substantial economic losses. In the US tens of millions of dollars are lost annually due to birds (King 2005) and similarly in Europe (IUCN 1997). The use of the robotic system offers fish farmers a continuous cost-effective and ecologically friendly bird deterrence sustainable solution. When birds land on the pond, they are scared away by a boat heading to their direction as quickly as possible, thus limiting the damage.

While ideally the input for the autonomous boat would come from computer vision algorithms analyzing the video collected from cameras placed around the pond, these capabilities have not matured enough and suffer from inherent limitations of separating the birds from the moving water background and ”guessing” bird's distance/location based on its size. Therefore the boat's navigation is being carried out using a human-sensor (preferably crowdworkers, due to their many advantages as enumerated at the beginning of the paper), based on a schematic pond map and the transfer of cameras output.

We use a web-based simulated version of the above system, which was primarily developed for experimentation.[2] In this system, new birds appear according to a pre-defined scenario which specifies their arrival time, specific location at the pond, and the time they will leave by themselves if not deterred by the boat by then. Birds are deterred whenever the boat gets near them (i.e., within some pre-set distance). To emphasize this capability the boat is enclosed with a circle representing its deterrence radius and once a bird is within the circle, it will be deterred immediately and will not be presented anymore (see Figure 1). Human sensors can see the pond area, the boat and arriving birds (in their proper location) through the system's GUI. They can convey this information to the system by clicking on the birds available (marking). The indication for a bird marked is a small blue circle next to it. If the bird leaves before deterred, the human sensor can ”unmark” it using a right-click. Any marking or unmarking event will activate the boat's route planning algorithm, possibly resulting a route update. In the absence of any markings, the boat stops, i.e., becomes idle. The joint goal (of the robot and the human sensor) is to deter as many birds as possible. Hence the measure of performance is the session score calculated as the number of birds deterred. Throughout the session, the system logs all marking and unmarking events (including the coordinates and timestamp), the current planned route (continuously), the boat's location (continuously), and the time and location of every bird deterring event.

The birds' deterrence testbed is a good representation of our problem domain: it contains a dynamic environment where new events of a spatial nature (represented by the birds that arrive and leave) need to be handled by a robot (the robotic boat), where crucial information that is unattainable by the robot (bird locations) can be easily obtained by

[2] The testbed was developed using IIS for the server-side and Angular framework (HTML, CSS, Typescript) for the client-side.

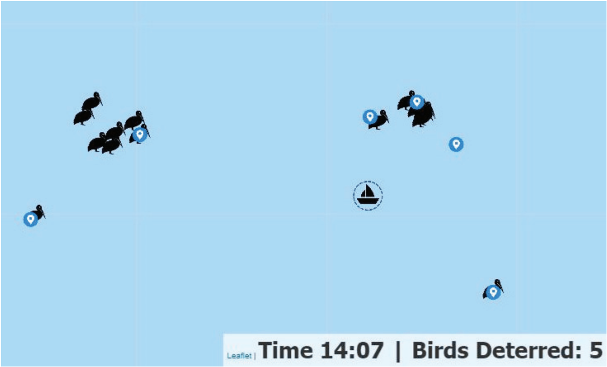

Figure 1: A Screenshot from the testbed used.

a crowdworker acting as a human sensor.

Experimental Design

We used the above testbed with a rectangular pond of size 500X1000 pixels. Session length was set to 15 minutes and the number of birds within was 750,[3] with random appearance times throughout. Birds leave pattern (i.e., if not deterred) was set such that at every second a bird is still present, it has a 4% chance to disappear. The robotic boat's speed was set to 62 pixels per second and the deterrence radius was set to 20 pixels. These parameters were defined as one of the representative realistic use cases in the larger bird-deterrence project. For route planning we used a greedy algorithm that always picks the minimum route among all routes starting from the current location and reaching two markings along the way. The algorithm is called on every new marking or unmarking of a bird. Subjects were recruited and interacted through AMT. Each participant received thorough instructions on how to use our framework in order to mark and unmark birds' location, the game rules and her goal in the game. This was followed by a short qualification quiz and short training to make sure she controls the marking technique. Compensation included a small show-up fee and a bonus which linearly depended on the number of deterred birds.

Results and Analysis

Overall we had 74 subjects initially requested to act as human sensors (denoted ”sensors” onward). As explained above, we base the performance of a sensor on the score she achieves, measured as the number of birds deterred by the robot. The average score obtained with the 74 subjects was 465. Since the total number of birds appearing in the experiment is 750, the average score obtained is equivalent to deterring 62% of the birds.

Performance Limits

While a 62% success might seem quite disappointing, we emphasize that deterring all 750 birds is practically unachievable. To better understand this point recall that the birds' deterrence success is influenced by three factors. The first is the completeness of the reporting made by subjects. The second is the effectiveness of the underlying route-planning algorithm used—in our case a greedy algorithm that is not necessarily optimal for this kind of online event allocation. Finally, even if continuously provided with complete information and using optimal routes, there is no guarantee all birds will be deterred, as it is possible that the pace of arriving birds, at times, is too high for the boat to reach them all, given its speed.

In order to properly reason about the influence of the above three factors, we present Figure 2, which unveils the performance limits of using a human sensor in our setting. Here, instead of using human subjects to report bird locations, we used a virtual sensor that provided the route planning algorithm with this information. The horizontal axis of the graph corresponds to the percentage of birds reported by the virtual sensor out of those present at any time and the vertical axis is the overall number of birds deterred.[4]

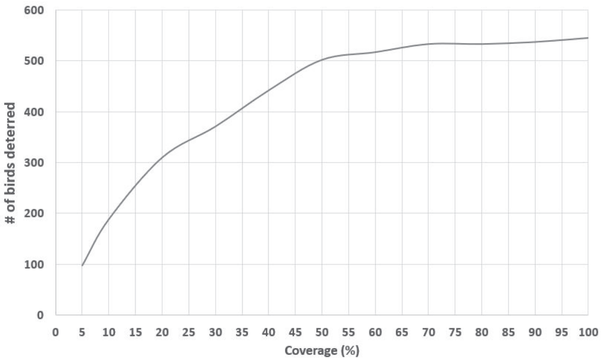

Figure 2: Score/Number of birds deterred as a function of the percentage of birds reported at any given time (i.e., coverage) by a virtual sensor.

We use the term ”coverage” to denote the percentage of birds reported by the sensor out of those present at any given time.[5] From Figure 2 we observe that, as one would expect, there is a strong (positive) correlation between performance and coverage. Still, even with 100% coverage the number of deterred birds is 545, rather than the theoretical bound of 750. Meaning that the navigation algorithm used and the environmental conditions (boat speed and deterrence radius, birds arrival rate and the time window they remain present) account to not deterring 205 birds.

Indeed, deterring 545 birds in our setting is an upper bound for human-sensor performance when taking the route planning algorithm to be given. Interestingly, even without the use of the sensor, the system can score quite substantially. For example, continuously traversing the pond area in

[3] Making it very hard for a human sensor to mark all birds, hence sensor's performance becomes an issue.

[4] The specific birds to be reported were randomly picked out of those currently present, according to the corresponding percentage on the horizontal axis.

[5] Relating only to birds actually present, i.e., excluding markings indicating a bird where there is really no birds present.

rectangular routes starting from its external perimeter and towards the center point (and back) yields a score of 286 (derived experimentally), which is more than we managed to achieve with the help of some of the sensors in our experiments.

Score Prediction Challenges

Naturally, different sensors achieve different scores. Figure 3 depicts the score of the 74 subjects in ascending order. It also includes the performance level achieved with the 100% coverage virtual-sensor and the performance the system may achieve even without using a human-sensor, as given above. Interestingly, we find that the performance with few sensors slightly exceeds the theoretical upper bound extracted with a 100% reporting sensor. For now, we defer the discussion on this phenomena. From the graph we observe that indeed there is a substantial variance between subjects in terms of the achieved score. Meaning that having the ability to predict individual score is highly advantageous—effective prediction will enable filtering subjects and assigning the task only to those who are likely to highly perform.

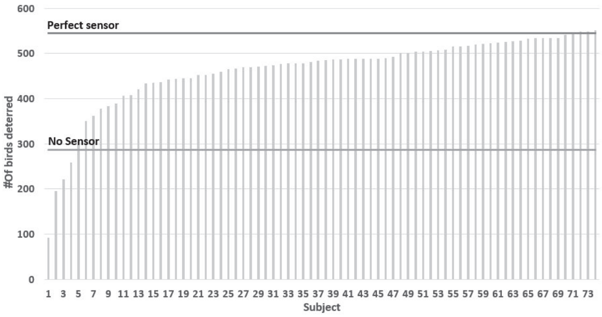

Figure 3: Score of each of the 74 subjects, ascending.

Predicting a new sensor's score in our case, however, is highly challenging for two primary reasons. First, the prediction should be carried out based on a short interval of operation. The greater the ratio between the total task's length and the amount of time used for prediction, the smaller is the overhead associated with the sensor's replacement. Furthermore, if the sensor turns to perform poorly, the longer the interval of operation used for prediction the greater the loss (in terms of score) due to this poor performance. Therefore an important prerequisite for effective prediction in the context of our model is that the performance of the sensor early in the task is strongly correlated with the overall performance throughout the session, as can be seen in table 1 The first column of the table is the length of the initial interval in the session (in minutes) for which we take score, and the second is the correlation with overall scores when considering all 74 sensors. The third and fourth columns correspond to the correlation obtained by omitting those 10% and 20% of the sensors that seem to have the weakest relationship between the score in their initial interval and the score in the full session, respectively. These latter two measures enable reasoning about the extent of influence exceptions may have on the obtained correlation. From the table we observe that indeed there is a strong correlation between the score achieved during the first few minutes of operation and overall score. In particular, a relatively high correlation is obtained even when considering only the first two minutes of operation. Therefore relying on the first two minutes of operation is a good basis for predicting the score for the entire session. Furthermore, the correlation does not seem to improve much with the filtering of individuals (as reflected in the third and fourth columns), providing further support for the potential of producing a rather accurate prediction based only on initial short experience with the sensor.

initial operation	r	r	r
(minutes)	**(all)**	**(90%)**	**(80%)**
1	0.43	0.5	0.5
2	0.74	0.82	0.87
3	0.82	0.84	0.88
4	0.89	0.90	0.92
5	0.93	0.94	0.95

Table 1: Correlation between the score achieved during the first $i \in \{1, 2, 3, 4, 5\}$ minutes of operation and the overall score in the task (i.e., over 15 minutes) for the 74 subjects

The second challenge in predicting the sensor's score throughout the session also results from the need to carry out the prediction in real-time. While the above-reported correlation results assure that the performance exhibited at early stages is correlated with the performance throughout the session, early-stage performance cannot be trivially measured. The system is unaware of the arriving birds nor their locations—perhaps these may be figured out in retrospect, however prediction is required in real-time to enable (and benefit from) sensor-switch. Meaning that the system can only base its prediction on measurable parameters that can be extracted from the initial interval, potentially characterizing the user's behavior throughout the session.

An important input the system can rely on for prediction is the markings made by the sensors. Indeed the system would not know if these are correlated with actual birds, whether there are other (unmarked) birds present and whether some of the already marked birds are no longer relevant as they have already left. Still, as we show later on, this information is invaluable for predicting performance. Other information includes marking-cancellations (overall and overtime), the number of seconds there are no markings available (hence the boat is idle), the distance between the new markings' locations and the boat's location at the time of marking, and length of the routes created by the route planning algorithm.

We use the above attributes (taken for the first two minutes of the session) as an input for prediction models developed with four standard Machine Learning (ML) methods: Linear Regression (Chatterjee and Hadi 2009), Random Forest (Segal 2003), Adaboost (Schapire 2013) and SGD (Bottou 2010). Indeed, these Machine Learning algorithms were developed for formal domains, which are materially different from ours where the goal is to predict human behavior (Shmueli 2017; Yahav, Shehory, and Schwartz 2018). Still, Recent research has found much merit in integrating data science and behavioral models in the context of predicting human choice behavior (Plonsky et al. 2017).

Our implementation used the scikit-learn Python package (Paper and Paper 2020), taking advantage if its GridSearchCv (Paper and Paper 2020) for tuning and parameter selection. The resulting models are based on 400 estimators and max depth of 8 with Random Forest, 400 estimators and a learning rate of 0.001 with Adaboost, and 1000 iterations and $\alpha = 0.0001$ with SGD.[6]

In order to evaluate the performance of the four models we randomly divided our data (traces of 74 sensors that took part in our experiments) to a training set (70%) and test set (30%), calculating the correlation (R^2) between the score prediction received and the actual score. This process was repeated 1000 times, taking the average of the R^2 values obtained with each prediction model. The results of the two highest-performing methods are summarized in (the second column of) Table2

Model	First Interval	All Intervals
Random Forest	0.15	0.19
AdaBoost	0.04	0.07

Table 2: Results of the highest performing models, for predicting the entire population's score

The third column of the table reports the performance of the same models, when the input is augmented in a way that training is carried out based on all two-minutes intervals spanning those sessions belonging to the training set (i.e., taking the first consecutive seven two-minutes intervals of the 15-minutes session, and their achieved score rather than merely the first interval). The advantage of this input augmentation method is that it substantially increases the size of the training set. On the other hand, the behaviors observed within the additional training data do not fully represent the behaviors to be observed in the first two minutes of operation of new sensors which performance needs to be predicted. This latter weakness finds some support from the non-perfect correlations found between the score in the first interval and the entire session, as in Table 1. Still, in many cases, as well as in our specific case, the advantages turn to be greater than the disadvantages. Alas, even with the slight improvement we obtain through input augmentation, we conclude, based on R^2 scores presented in the Table 2, that the performance prediction with all models is quite poor.

Sensors versus Operators

The relative failure of standard machine learning methods to predict sensors' performance possibly suggests that there is some inherent difference between different sensors' marking strategy which precludes the identification of a general pattern. To investigate this hypothesis we turn back to a more detailed analysis of the coverage parameter that was proven instrumental in its influence over performance when experimenting with the virtual sensor (see Figure 2). One interesting observation that can be drawn from the figure is that the average performance obtained with human sensors (465 deterred birds) is equivalent to having the virtual sensor continuously report 44% of the birds that are present at any moment. However, checking the actual coverage obtained in the 74 sessions, we find that the average coverage was 36%. This significant difference can have two possible explanations. The first is that there is a substantial variance in individual coverage, hence a simple average is not indicative. The second explanation is that some people's markings are strategic, meaning that they aim to influence (and possibly optimize) the boat's route. Indeed the variance in coverage is 498 (standard deviation is 22.3), supporting the first explanation. For the second explanation we introduce Figure 4 which depicts the relationship between coverage and score—each data point in the graph represents the coverage (horizontal axis) and score (vertical axis) obtained by one of the 74 sensors. From the graph we observe that there is a poor correlation between coverage and score ($r = 0.25$). The graph also includes the curve representing the score as a function of coverage obtained with a virtual sensor (taken from Figure 2). Indeed, the performance of some of the sensors adheres to this curve, while others are very far from it. Meaning that those latter subjects are not functioning as a sensor in the sense of reporting everything (or a random portion of what) they see. Instead they are applying some logic in deciding which birds to mark. This external logic that guides the choice of which birds to mark accounts, at times, to an improvement in the achieved score compared to the score with a virtual sensor associated with the same coverage level—Indeed, some of the sensors managed to achieve a score greater than the score of a virtual sensor that reports all birds. Many others have managed to achieve a score that is more than twice the score of a virtual sensor associated with a similar coverage level.

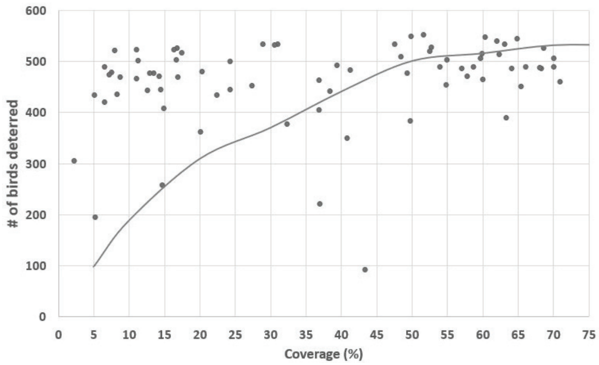

Figure 4: Score according to coverage percentage.

One interesting group of sensors emerging from the graph is of those associated with coverage smaller than 33%, and score higher than 400 (30 out the 74 subjects). Sensors in this group are characterized by a very low coverage and yet the great majority of these, manage to achieve a score substantially greater than the expectations based on a virtual sensor. In fact, the average score of the sensors from

[6] The rest of the parameters in each model are use the scikit-learn defaults).

this group does not fall short from the average of all sensors with greater (2-4 times) coverage. In order to better characterize the behavior pattern of sensors of the latter group, we went over the log files, replicating all sessions, manually exploring the reflected behaviors. Doing so, we found a recurring pattern - sensors from this group repeatedly mark 1-3 birds at a time, wait for the boat to visit these locations, then wait a few seconds, mark again 1-3 birds and so on. Marking 1-3 birds at a time leave the boat route planning module very little flexibility, and in most cases dictates one or two possible routes. Meaning that the sensor is actually controlling the boat's route through her markings. One reasonable explanation for the repeated few seconds waiting before marking additional birds is that the sensor is engaged in some planning related to the next markings. Hence it is this planning that accounts for the score performance those subjects achieve, as even though these sensors provide the system with very little information, the simple routes produced based on this information turn to be highly efficient, in terms of the scored achieved. Evidence for people's tendency to be in control can be found in various other general literature from psychology and behavioral economics (Adler 1930; Burger 1985). Therefore, we distinguish the sensors aiming to control the boat's route through their markings (characterized by $coverage < 33\%$ and $score > 400$), denoting them ”operators” onward. We keep referring to all non-operators human-sensors as ”sensors”. While we do not expect any correlation between score and coverage within the operators group, we do expect that correlation within the sensors group to substantially improve upon excluding the operators. Indeed once excluding operators, the correlation is $r = 0.59$ (compared to 0.25, when taking all subjects).[7]

Performance Prediction with Operators Distinction

The distinction between operators and sensors carries much potential to improve the poor results obtained by prediction models constructed based on the entire population. If we can effectively predict the score for individuals from each group and have the ability to classify subjects to the different groups in runtime, then overall performance can be substantially improved by dropping those with low predicted performance early in the session. Therefore we now turn to show that based on a short initial interval of operation we can: (a) effectively distinguish operators from sensors; (b) predict the score for sensors; and (c) predict the score for operators.

Distinguishing between Sensors and Operators. As established above, the primary measures of differentiating sensors from operators are the coverage and score. Unfortunately, these measures can not be computed by the system in real-time as we do not have the total number of birds available at any given time, nor their locations. Our classification model thus relies on the measurable behavioral features mentioned earlier in this section, in particular on the number of clicks (both markings and unmarkings), the number of marking cancellations, and the number of seconds the boat was idle (i.e., having no bird markings to plan over). We use the same ML methods as above (excluding Linear Regression which is irrelevant for our classification task), once again randomly dividing the 74 sessions to a training set and a test set, in a ratio 70%-30%, repeating the process 1000 times and averaging the resulting R^2 value. Table 3 provides the average accuracy percentage (i.e., the percentage of accurate classifications for sensors of the test set).

Model	First Interval	All Intervals
Random Forest	81.67%	86.1%
AdaBoost	67.6%	81.8%
SGD	72.76%	74.3%

Table 3: Average accuracy percentage of separation between sensors and operators

The second column provides the results obtained when relying solely on the first two minutes of each session, and the third column is for the prediction achieved with the input augmentation method, as used above. We observe that the classification accuracy achieved with all three methods is quite high, especially when using the augmented input. Meaning that upon the recruitment of a new sensor, we can reliably classify her (with 86% accuracy) as an operator or sensor based on her first two minutes of operation.

Predicting Sensors Performance. Non-operator sensors' performance prediction can be based on the behavior observed during the same two-minutes interval used for the classification. We experimented with the same four prediction models as above, using the same methodology for randomly dividing sessions into a training set and a test set repeating the process 1000 times. This time, the behavioral features found to be most effective for prediction were the number of clicks overall (markings and unmarkings), number of marking cancellations, total idle time, the average number of markings present throughout the session and the average distance between marking and the boat. Table 4 provides the R^2 values obtained with the two highest-performing prediction models:

Model	First Interval	All Intervals
Random Forest	0.41	0.58
AdaBoost	-0.116	0.2707

Table 4: Results of the highest performing models, for predicting sensor's score

The third column in the table presents the results obtained with input augmentation as before. The results reflect a relatively high prediction accuracy with Random Forest, in particular with input augmentation (an increase to 0.58, from

[7] We do not expect this correlation to be close to 1, because, as explained above, even in this group some of the subjects apply some human intelligence when prioritizing the birds to be marked, given that they do not have the capacity to mark all of them. This is evidenced by the fact that some of them managed to perform close to 100% marking with virtual sensor even though their coverage is lower than 100%.

0.1 9 with a model trained over all sensors, i.e., without excluding operators). Interestingly, the increase in performance occurs despite the decrease in the training set size, as upon excluding operators we are left with a substantially smaller population of sensors.

Predicting Operators Performance. Predicting the performance of human-sensors classified as operators is way more challenging than predicting the performance of "regular" sensors. This is simply because, as discussed above, there is no correlation between score and coverage within the operators group. Still, we manage to develop a prediction model based on the number of clicks (markings and unmarkings) and the number of marker cancellations over the first two minutes of operation that yield a R^2 value of 0.19 with Linear Regression, as detailed in Table 5, reporting the results of the two highest-performing methods:

Model	First Interval	All Intervals
Linear Regression	0.19	0.13
Random Forest	0.07	0.1

Table 5: Results of the highest performing models, for predicting operator's score

This is the same level of accuracy obtained when predicting based on the general population, as reported above. Still, given the limitations of prediction for this population as discussed above, and the decrease in the test set size, as we rely only on operators, this result is quite satisfying.

Prediction-Based Sensor Switching

We turn to illustrate the benefit of using the above classification and prediction models for dynamic sensor assignment. For this, we consider a setting where the boat is operated continuously for a very long time. Sensors can be recruited for 15 minutes, yet can be replaced at any time. Therefore, we continuously iterate over the 74 sensors used in our experiments, and determine for each, after two minutes of operation, whether to keep her for an additional 13 minutes or replace her with the next sensor in line (whose identity and performance are unknown). The decision-making rule associated with replacing a sensor is based on a performance threshold r. Meaning that we replace the sensor only if her predicted performance is lower than r. To make the most use of the 74 sensors' data, we train the classification and prediction models used with each sensor from scratch, based on the data of the 73 other sensors.

Figure 5 depicts the average score per 15 minutes obtained in the above sensor dynamic assignment setting, as a function of the threshold r used (horizontal axis).[8] Each curve corresponds to a different approach used: (a) "no switching" - having all 74 sensors perform without switching (i.e., performance is the average of individual scores); (b) "unified population" - when score prediction models are based on all other 73 sensors; (c) "operator/sensor distinction" - when score prediction is based on preliminary classification as operator or a sensor, applying the relevant prediction model according to the classification; and (d) "perfect prediction" - knowing the real score of that sensor. As expected, the general behavior of all prediction-based curves suggests an increase as the threshold used increases. From some point, however, performance decreases with the increase in the threshold used. This happens for two reasons. First, while only those sensors with a (predicted) performance greater than r are used for full 15 minutes, each worker replaced contributes her own two-minutes score to the general score. As r increases, more workers are being replaced, and the relative weight of their two-minutes score contribution in the overall score becomes more substantial. Second, since prediction is not perfect, for high r values the relative weight of the 15-minutes score contribution of sensors identified as highly-performing while their real score is actually poor becomes more substantial.

The "perfect prediction" and "no switching" curves convey the range of improvement any prediction method can potentially achieve. With no prediction, the average score is 465 whereas perfect prediction obtains 511. When the prediction is based on the entire sensors population, the maximum score achieved is 478. Meaning that we manage to overcome 28% of the gap between no prediction and perfect prediction. With our proposed design that employs prediction based on sensor/operator classification, we manage to achieve a score of 496, i.e, overcoming 67% of the gap. Though despite the fact that each of the individual prediction models for the two populations is trained over a substantially smaller training set, the results obtained are more than twice as good.

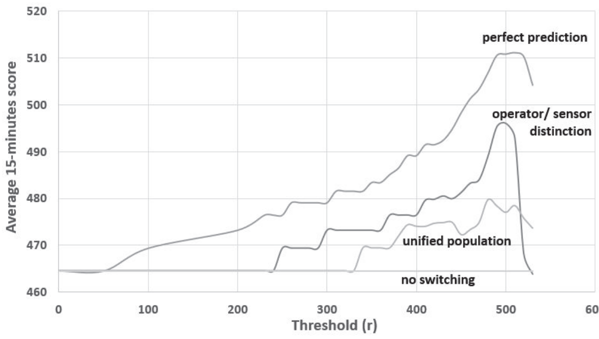

Figure 5: Average 15-minutes score in the sensor dynamic assignment setting, using the different prediction methods, as a function of the threshold r used.

Discussion and Conclusions

The encouraging results reported in the former section suggest that the prediction of crowdworkers' performance when acting as a human-sensor for a robot in search and rescue-like settings can be substantially improved by applying a preliminary classification of workers and developing a separate prediction model for each class. Failing to do so, re-

[8] The calculation averages all actual scores of the 74 sensors, either per two minutes or 15 minutes of operation, based on whether replaced or not, weighted accordingly.

sults in poor performance as the unified prediction model is strongly affected by those who attempt to influence the route planning process through selective reporting. The ability to effectively predict a human-sensor performance has many applications, among which the dynamic assignment (and replacement) of workers throughout the task, as demonstrated above. Predicting operators' performance is highly challenging, as their performance derives from the quality of the internal planning they apply, rather than the amount of event-related information they provide. Indeed, for those acting as classic sensors a way more accurate prediction was derived. Still, it is with this population that accurate prediction accounts most—any future improvement in the route-planning module will increase the average score of workers from this population, whereas will not affect the score of operators (as the robot's route-planning becomes simple given the small number of events they report at any time). Meaning that with a highly effective route planning module, the optimal assignment procedure would be to replace all those classified as operators along with those classified as sensors with poor score prediction.

The question of why some workers act as operators rather than sensors is intriguing. Alongside the behavioral explanation provided in this paper, we note secondary reasons such as misunderstanding the instructions (despite succeeding in the qualification quiz in the experiments and the pre-experiment simulation). Other reasons include our non-optimal route planning algorithm that might have pushed participants to think they can perform better by affecting navigation, and the rewarding mechanism that was based solely on score rather than (at least partially on) the ability to follow task instructions. Another conceptual question raised is the realism of using crowdworkers for high-risk search and rescue missions, in the terms of the responsibility for the consequences of non-optimal operation by crowdworkers for many reasons.

One limitation of our work is that the experiments were conducted using only one set of parameters. Experimenting with additional scenarios and possibly extending the number of subjects in each experiment will extend the applicability and the ability to generalize to other domains, most importantly perhaps is the search-and-rescue field. An additional direction for future work is the design of designated interfaces for those classified as operators. Meaning that upon identification as an operator, the worker will be requested to actually operate the boat rather than keep providing information about events. This may improve the score of that population.

Acknowledgements

This research was partially supported by the ISRAEL SCIENCE FOUNDATION (grant No. 1162/17),
and the ISRAELI MINISTRY OF SCIENCE & TECHNOLOGY (grant No. 89583)

References

Adler, A. 1930. Individual psychology. In *Psychologies of 1930.* Clark University Press. 395–405.

Alexander, G.; Raja, A.; and Musliner, D. J. 2008. Controlling deliberation in a markov decision process-based agent. In *Proceedings of AAAMS-Volume 1*, 461–468.

Almosalami, A.; Jones, A.; Tipparach, S.; Leier, K.; and Peterson, R. 2018. Beachbot: Crowdsourcing garbage collection with amphibious robot network. In *ACM*, 333–334.

Apostolopoulos, D.; Pedersen, L.; Shamah, B.; Shillcutt, K.; Wagner, M.; and Whittaker, W. 2001. Robotic antarctic meteorite search: Outcomes. In *ICRA'01*, 4174–4179.

Bottou, L. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*. Springer. 177–186.

Breazeal, C.; DePalma, N.; Orkin, J.; Chernova, S.; and Jung, M. 2013. Crowdsourcing human-robot interaction: New methods and system evaluation in a public environment. *Journal of Human-Robot Interaction* 2(1):82–111.

Bruemmer, D. J.; Boring, R. L.; Few, D. A.; Marble, J. L.; and Walton, M. C. 2004. " i call shotgun!": an evaluation of mixed-initiative control for novice users of a search and rescue robot. In *Systems, Man and Cybernetics*, volume 3, 2847–2852.

Burger, J. M. 1985. Desire for control and achievement-related behaviors. *Journal of Personality and Social Psychology* 48(6):1520–1533.

Chatterjee, S., and Hadi, A. S. 2009. *Sensitivity analysis in linear regression*, volume 327. John Wiley & Sons.

Chung, J. J. Y.; Xiao, F.; Recker, N.; Barnes, K.; Banovic, N.; and Lasecki, W. S. 2019. Accident prevention with predictive instantaneous crowdsourcing. In *CHI*.

Dai, P.; Rzeszotarski, J. M.; Paritosh, P.; and Chi, E. H. 2015. And now for something completely different: Improving crowdsourcing workflows with micro-diversions. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, 628–638.

Diamantas, S. 2020. Towards resolving the kidnapped robot problem: Topological localization from crowdsourcing and georeferenced images. In Arai, K., and Kapoor, S., eds., *Advances in Computer Vision*, 551–563. Cham: Springer International Publishing.

Dorais, G., and Kortenkamp, D. 2000. Designing human-centered autonomous agents. In *Pacific Rim International Conference on Artificial Intelligence*, 321–324. Springer.

Elmalech, A.; Sarne, D.; David, E.; and Hajaj, C. 2016. Extending workers' attention span through dummy events. In *AAAI Conference on Human Computation and Crowdsourcing*, 42–51.

Endsley, M. R. 2016. *Designing for situation awareness: An approach to user-centered design.* CRC press.

Faessler, M.; Fontana, F.; Forster, C.; Mueggler, E.; Pizzoli, M.; and Scaramuzza, D. 2015. Autonomous, vision-based flight and live dense 3d mapping with a quadrotor micro aerial vehicle. *Journal of Field Robotics* 33(4):431–450.

Ghandour, M.; Liu, H.; Stoll, N.; and Thurow, K. 2016. A hybrid collision avoidance system for indoor mobile robots based on human-robot interaction. In *Mechatronika (ME)*, 1–7.

Goodrich, M. A.; Schultz, A. C.; et al. 2008. Human–robot interaction: a survey. *Foundations and Trends® in Human–Computer Interaction* 1(3):203–275.

Ito, K., and Maruyama, H. 2016. Semi-autonomous serially connected multi-crawler robot for search and rescue. *Advanced Robotics* 30(7):489–503.

IUCN, E. P. 1997. The negative impact of birds on fish ponds. *Fishing for a Living: The Ecology and Economics of Fishponds in Central Europe* 79–81.

Jones, J. L.; Mack, N. E.; Nugent, D. M.; and Sandin, P. E. 2019. Autonomous floor-cleaning robot. US Patent 10,433,692.

King, D. T. 2005. Interactions between the american white pelican and aquaculture in the southeastern united states: an overview. *Waterbirds* 28(sp1):83–86.

Kitano, H.; Asada, M.; Kuniyoshi, Y.; Noda, I.; and Osawa, E. 1997. Robocup: The robot world cup initiative. In *AAMAS*, 340–347.

Kosti, S.; Sarne, D.; and Kaminka, G. A. 2014. A novel user-guided interface for robot search. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 3305–3310.

Krueger, G. P. 1989. Sustained work, fatigue, sleep loss and performance: A review of the issues. *Work & Stress* 3(2):129–141.

Lewis, M.; Wang, H.; Velagapudi, P.; Scerri, P.; and Sycara, K. 2009. Using humans as sensors in robotic search. In *2009 12th International Conference on Information Fusion*, 1249 – 1256.

Lewis, M.; Sycara, K.; and Nourbakhsh, I. 2019. Developing a testbed for studying human-robot interaction in urban search and rescue. In *HCII'03*, 270–274.

Lösch, R.; Grehl, S.; Donner, M.; Buhl, C.; and Jung, B. 2018. Design of an autonomous robot for mapping, navigation, and manipulation in underground mines. In *IROS*, 1407–1412.

Mason, W., and Suri, S. 2012. Conducting behavioral research on amazon's mechanical turk. *Behavior research methods* 44(1):1–23.

Moradi, M.; Ardestani, M. A.; and Moradi, M. 2016. Learning decision making for soccer robots: A crowdsourcing-based approach. In *IRANOPEN*, 25–29.

Mostafa, S. A.; Ahmad, M. S.; and Mustapha, A. 2019. Adjustable autonomy: A systematic literature review. *Artif. Intell. Rev.* 51(2):149–186.

Nimni, I., and Sarne, D. 2020. Effective operator summaries extraction. In *Proceedings of AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. forthcoming.

Nourbakhsh, I.; Sycara, K.; Koes, M.; Yong, M.; Lewis, M.; and Burion, S. 2005. Human-robot teaming for search and rescue. *Pervasive Computing, IEEE* 4(1):72–79.

Paper, D., and Paper, D. 2020. Scikit-learn classifier tuning from complex training sets. *Hands-on Scikit-Learn for Machine Learning Applications: Data Science Fundamentals with Python* 165–188.

Plonsky, O.; Erev, I.; Hazan, T.; and Tennenholtz, M. 2017. Psychological forest: Predicting human behavior. In *AAAI*, 656–662.

Pollack, M. E.; Tsamardinos, I.; and Horty, J. F. 1999. Adjustable autonomy for a plan management agent. In *AAAI Spring Symposium on Agents with Adjustable Autonomy*, 22–24. Citeseer.

Rahmawati, E.; Sucahyo, I.; Asnawi, A.; Faris, M.; Taqwim, M. A.; and Mahendra, D. 2019. A water surface cleaning robot. *Journal of Physics: Conference Series* 1417:012006.

Sampedro, C.; Rodriguez-Ramos, A.; Bavle, H.; Carrio, A.; de la Puente, P.; and Campoy, P. 2019. A fully-autonomous aerial robot for search and rescue applications in indoor environments using learning-based techniques. *J. Intell. Robotic Syst.* 95(2):601–627.

Scerri, P.; Pynadath, D. V.; and Tambe, M. 2002. Towards adjustable autonomy for the real world. *Journal of Artificial Intelligence Research* 17:171–228.

Schapire, R. E. 2013. Explaining adaboost. In *Empirical inference*. Springer. 37–52.

Segal, M. 2003. Machine learning benchmarks and random forest regression. *Technical Report, Center for Bioinformatics & Molecular Biostatistics, University of California, San Francisco*.

Sheridan, T. B. 2016. Human–robot interaction: status and challenges. *Human factors* 58(4):525–532.

Shmueli, G. 2017. Analyzing behavioral big data: methodological, practical, ethical, and moral issues. *Quality Engineering* 29(1):57–74.

Tomko, M., and Richter, K.-F. 2015. Defensive wayfinding: Incongruent information in route following. In Fabrikant, S. I.; Raubal, M.; Bertolotto, M.; Davies, C.; Freundschuh, S.; and Bell, S., eds., *Spatial Information Theory*, 426–446. Cham: Springer International Publishing.

Wang, H.; Kolling, A.; Brooks, N.; Owens, S.; Abedin, S.; Scerri, P.; Lee, P.-j.; Chien, S.-Y.; Lewis, M.; and Sycara, K. 2011. Scalable target detection for large robot teams. In *HRI'11*, 363–370.

Winter, S.; Tomko, M.; Vasardani, M.; Richter, K.-F.; and Koshelham, K. 2017. Indoor localization and navigation independent of sensor based technologies. Technical Report 9(1), Umeå University, Department of Computing Science.

Yahav, I.; Shehory, O.; and Schwartz, D. 2018. Comments mining with tf-idf: the inherent bias and its removal. *TKDE* 31(3):437–450.

Zhao, L.; Lawhorn, R.; Wang, C.; Lu, L.; and Ouyang, B. 2019. Synthesis of robot hand skills powered by crowdsourced learning. In *ICM*, volume 1, 211–216. IEEE.

Zilberstein, S. 2015. Building strong semi-autonomous systems. In *AAAI*, 4088–4092.