Factoring the human decision-making limitations in mobile crowdsensing

Merkouris Karaliopoulos*, Leonidas Spiliopoulos† Iordanis Koutsopoulos*
*Department of Informatics, Athens University of Economics & Business, Athens, Greece, Email: {mkaralio.jordan}@aueb.gr
†Max Planck Institute for Human Development, Berlin, Germany, Email: spiliopoulos@mpib-berlin.mpg.de

Abstract—In this paper we address the way crowds of mobile end users are matched with crowdsensing (MCS) tasks. We challenge two practices/assumptions that are dominant in the related literature. The first one consists in approaching the problem as an instance of optimal centralized assignment of users to tasks. We argue instead that it is more plausible to view this as a task selection problem on the user side, i.e., users select tasks, and cast it as a multi-attribute decision problem with many alternatives. The second monotonously repeated assumption confronted in this paper is that end users, either when bidding for a specific task or in the few cases that themselves select tasks, behave as fully rational agents (strategically) seeking to make optimal choices. We rather explicitly acknowledge the bounded rationality of human decision making as this results from the human cognition processes and their inherent limitations.

We first summarize long research threads on cognitive decision-making in the fields of economics and psychology and iterate on models (heuristics) that mobile end users possibly activate when selecting of crowdsensing tasks. Then, we describe what these heuristics imply for the end-user characterization (profiling) process and how their activation can be inferred out of past choices. Finally, we draw on empirical data collected through an online questionnaire to conduct a comparative study of different models for MCS task selection.

I. INTRODUCTION

Mobile crowdsensing (MCS) has emerged over the last decade as a powerful service paradigm promising to transform the way information is generated and disseminated [15] [35]. Multiple types of data (photos, video clips, temperature/humidity measurements and positioning data) can be (semi-)automatically generated by the various sensors embedded in current and forthcoming smart devices, with little or no human intervention. At the same time, a variety of social mobile applications has made the uploading and sharing of this information easier than ever before. Most importantly, these applications aggregate and process information contributed by the end users in various ways and generate collective knowledge/awareness about a phenomenon or condition of interest. This knowledge can be of value to third parties, who may build or subscribe to a commercial service on top of it, but may also be fed back to the data contributors themselves as a community service.

The applications that leverage mobile crowdsensing are diverse in scope and objectives. Among the most popular ones count those monitoring environmental conditions such as air pollution, humidity, and electromagnetic field (EMF) radiation; those tracking the levels of urban road traffic and parking space availability [1] [2]; as well as applications that report on events as they happen (social journalism) or inform (semi-)real-time about the availability of purchase opportunities, e.g., how may PCs on sale are left in a particular store [4].

In general, we can distinguish four main parts in a MCS system. The first one is the crowd of mobile users who contribute data to the system (data contributors) via the increasingly rich sets of embedded sensors in their smart devices. Both the type and volume of the contributed data may vary across different applications, from a couple of bytes for measurement data up to megabyte pictures captured with the cameras of the smart devices.

Crowdsensed data are typically submitted to a mobile platform via a client application running on users’ devices. The functionality and complexity of these platforms are also variable and, most often, positively correlated. Often running in a cloud, the platform notifies potential contributors about different ongoing crowdsensing tasks, recruits them for tasks they have expressed their interest in, and collects their contributions. It may also implement mechanisms for motivating end users to contribute to crowdsensing tasks.

The third part of an MCS system is the Service Provider (SP), i.e., the entity that runs the crowdsensing campaign, processes the crowd data and makes up a service out of them. Generally, the SP entity may or may not coincide with the platform owner.

Finally, (data consumers or service subscribers) make use of the service that is offered by the SP out of the crowdsensed data. The sets of data contributors and data consumers may be disjoint or not. For example, consider the case where an SP recruits mobile users to report, through text or pictures, the status of automatic vendor machines across the city. The data consumer then is a commercial entity, the owner of automatic vendor machines. On the other hand, there are many applications where these two sets partially or fully overlap. This is the case, for instance, with Waze [1] or social parking applications [3] [2], whereby the mobile end users may exchange roles over time. They serve as contributors of information about traffic conditions and parking availability, respectively, at one point in time; while, they may benefit from such information contributed by other users, at another.

In this paper we address the way data contributors end up carrying out specific crowdsensing tasks. Sensor recruitment/selection and task allocation/assignment are the most frequently used terms in literature to point to this process;
both reflect the strong bias towards solutions that involve centralized decisions on the platform/SP side.

A. Existing assumptions and related work

In the vast majority of literature, the crowdsensing tasks are centrally assigned to rather than selected by the end user (e.g., in [31] [18] [30]. The end user’s preferences and skills are most often abstracted and hidden into a payment that she requests in order to participate in a task. Implicitly, the user will ask more for a task that does not fall within her interests and may demand more time and skills to carry out (e.g., taking a few photos from a coffee place). On the other hand, the tasks are linked to physical locations and the distance that needs to be traversed by users sets a bound on how many contributions they can make. Then a centralized entity (platform), which is aware of all data contributors and ongoing tasks, assigns data contributors to tasks by solving a constrained optimization problem, whose details vary from work to work.

For instance, in the simpler case that the task distance-related constraints are not taken into account and each task has its own finite budget, the sensor recruitment process yields an instance of the budgeted maximum coverage problem [31], which can be solved with the algorithm specified in [24]. In the other two studies, the assignments of contributors are made simultaneously to all tasks. Hence, in [18] a net reward is made by the platform (service provider) if contributor $u$ carries out task $l$. On the problem constraints’ side, contributors have time budgets and tasks need to attract some predefined number of contributors. The platform’s optimization objective is then to maximize its aggregate reward over all tasks and users assigning them in a way that satisfies the users’ time budget and the tasks’ coverage constraints. Very similar constraints are considered in [30]; only, in that case, the cost of a contributor-to-task assignment coincides with the Euclidean distance between the two and the optimization objective is the minimization of the aggregate cost over all contributor-to-task assignments. The optimization is carried out in two steps aiming at preserving the location privacy of the contributors: the first step is carried out centrally with obfuscated contributors’ locations as inputs, and the second one locally at their devices.

Much fewer are the studies that place the burden of decisions on the contributors. In [12] crowdsensing tasks also have deadlines, within which they need to be carried out. Each user then independently seeks to derive the longest sequence of task locations that she can sequentially visit without violating those deadlines. In [12] the contributors do not receive payments; they rather make themselves available as volunteers towards a common purpose, the fulfillment of as many tasks as possible. Yet the contributors are assumed to be optimizers in that they actively look for the maximum possible sequence of tasks they can serve. An even more demanding model for contributors is set forth in [25], where the optimal reverse auction is sought for under the assumption that the contributors are thinking strategically in declaring their bids.

B. Our contribution

We challenge the existing assumptions in the way data contributors and crowdsensing tasks are matched in two ways.

First of all, crowdsensing tasks are not centrally assigned to the end-users. Rather end users select which crowdsensing tasks they will contribute to out of a set of relevant alternatives. This ultimate freedom of choice is certainly desirable by end users, as repeatedly reported in interview-based studies of mobile crowdsourcing applications (for example, [4] [36]) and may even be unavoidable in view of the highly varying privacy concerns of end users.

Secondly, and most significantly, end users exhibit what is called bounded rationality. Their decisions are subject to human cognitive limitations and biases [20] and seek to satisfice rather than optimize. Satisficing, a portmanteau word of satisfy and suffice, implies searching through the available alternatives until one is deemed acceptable. It was used by Herbert A. Simon in [33] to describe the behavior of decision makers given that they can rarely know and evaluate all possible outcomes of their decisions with sufficient precision due to limited memory and processing capacities.

We will show that this setup has consequences for the profiling of users, which is now carried out over models bearing strong structure, i.e., explicit theory for the decision-making process that interprets into specific parametric form and relations between the decision variables. This changes dramatically the way the model parameters are inferred when compared to standard structure-agnostic and atheoretical machine-learning techniques that are currently dominating the behavior modeling task.

We formalize the MCS task selection problem faced by to-be-task-contributors in Section II. Section III then presents cognitive heuristic models, proposed in the fields of economy and psychology, that capture the bounded rationality of human decision-making and could guide users’ choices of MCS tasks. In Section IV, we demonstrate how users can be profiled in light of these models and draw on empirical data from an online questionnaire to compare different decision-making heuristics for MCS task selection.

II. THE SYSTEM MODEL

Formally, let $U$ be the set of potential data contributors. This consists of mobile users with smart devices who have registered with an MCS platform and run a client application on their device. Let also $L_i(u), u \in U$ be the set of crowdsensing tasks each user is presented with as she roams in the city, on her way to work, back home or during her leisure time. The set of tasks varies with time depending on the user’s location and, possibly, other context information that is collected by the application running on the user’s smart device.

Each task $l \in L_i(u)$ can be characterized by a set of attributes $a(l) = (a_1, a_2, ..., a_n)$. These include the reward, say $a_1$, that a task may offer to data contributors; the physical distance of the contributor from the location related to the task or the required deviation from her nominal route in order to carry out the task; the battery consumption related to the task; and the context (e.g., commercial or non-profit) of the underlying service that is facilitated with the contributed data. Each of these attributes (interchangeably called cues hereafter) induces different orderings of $L_i(u)$, which, in turn,
vary across the data contributors, depending on their interests, preferences, and pro-social attitude.

Hence, each user is faced with a *multi-attribute choice problem with many alternatives* (tasks). However, we need to distinguish between two general scenarios, dictated by how the tasks reward their contributors. There are tasks that reserve rewards as soon as the mobile user expresses her intention to contribute to them. Namely, the task is advertised by the application/platform only as long as there is budget left for it. In that case, each user independently decides whether to undertake the task with the specific fixed reward. If she does, she deterministically receives the advertised reward. On the other hand, there are tasks that induce competition among the contributors and relate their rewards to the number of contributors who will emerge for the task. “Pay the first $k$ a fixed reward” or “Divide equally the task budget among those who will make a contribution within time $T$” are examples of reward policies that create dependencies between the decisions of users and motivate them to think and act strategically. Likewise, dependencies between the users’ decisions, are generated by policies of the type “a reward is given only when a minimum of $k$ contributions are made”, this time motivating the coordination rather than the competition of end users.

More formally for a task $l$, let $c^l_u$ be the decision/action of contributor $u$ and $c^l_{-u}$ be the set of decisions of other contributors. Then the contributor’s $u$ reward for the task $l$ depends on the task. For individual decision-making:

$$a_1 = f(c^l_u) \forall u \in U$$  \hspace{1cm} (1)

For strategic decision-making:

$$a_1 = f(c^l_u, c^l_{-u})$$  \hspace{1cm} (2)

### III. Models of Roundedly Rational Decision-Making

We know from the behavioral sciences that the decisions of end users deviate considerably from those of a perfectly rational entity. Human rationality is bounded by cognitive constraints, such as memory and information processing capacity, as argued already in the mid sixties by Herbert Simon [32] [33] and demonstrated in a number of laboratory experiments (e.g., [17] [5]). Also, individual heterogeneity in these constraints, and ultimately behavior, is widespread across all types of decision-making tasks. It is well known that individual heterogeneity must be respected in cognitive modelling, otherwise significant biases and erroneous conclusions can be drawn from pooling data together and estimating a “representative” decision-maker [37].

Models of bounded rationality have been largely investigated in the fields of cognitive psychology and behavioral economics, often with the use of different techniques. An important distinction is made between individual decision-making (iDM) and strategic decision-making (sDM) tasks. In iDM, the outcomes for a decision maker depend only on her own choices and possibly probabilistic draws from nature (see eq. 1). In sDM, the outcomes depend not only on own choices but also the choices of others (see eq. 2)—this is the quintessential problem of strategic interactions that game theory addresses. Different models of choice have been proposed for these two types of tasks.

Psychologists’ attempts to flesh out Simon’s vision and construct models of the heuristics people use to reason have led to two different viewpoints. The first one is most famously expressed in the “heuristics and biases” program of Kahneman, Tversky and colleagues [21]; the second one in the “fast and frugal heuristics” program of Gigerenzer and colleagues [16]. The former program focused on using laboratory experiments, typically with students as subjects, to document deviations of human behavior from the neoclassical economic model. These deviations were then described with verbal labels such as “availability bias” and “representativeness bias”. The latter program focused on developing a suite of simple mathematical models known as simple decision heuristics, and investigating how well these models describe human behavior and how well they perform in real-world problems. These fast and frugal heuristics (i) rely heavily on core human capacities (such as memory recognition and recall); (ii) do not necessarily use all available information, and process the information they use by simple computations (such as using only one piece of information); (iii) are easy to understand, apply, and explain. Surprisingly, these heuristics can outperform computationally more complex methods developed in statistics and artificial intelligence [22]. Thus heuristics may well be closer to the unknown optimum, at least under some conditions.

#### A. Individual decision-making models

1) Decision making under certainty: An important family of fast and frugal heuristics are the lexicographic heuristics, whereby the user inspects the attributes in a specified order and makes a decision based on the first attribute that discriminates between her alternatives. Notably, the user does not process all information at hand (i.e., full list of attributes) to make a choice. The performance of lexicographic heuristics, in terms of accuracy or utility maximization, has been studied via computer simulations and mathematical analysis in datasets from business, medicine and psychology. Overall, three major stylized facts have emerged ([22]). First, there are small differences in performance between heuristics and more complex benchmarks such as linear regression, neural networks, classification and regression trees or naïve Bayes. Second, simple heuristics often have higher performance in out-of-sample prediction. Third, each heuristic or benchmark outperform the other under certain conditions, since each one is capable of exploiting different properties of an environment. The coupling of the structure of decision rules and the environment allows the former to efficiently exploit the latter, in what is called ecological rationality.

The LEX and LEX-Semi heuristics: Lexicographic heuristics may operate over binary or continuous cues and be used to select between two or more alternatives. The most straightforward heuristic in this category, called LEX in literature [14], works with continuous cues. It inspects the cues in order of decreasing importance, and stops upon the first cue that discriminates between the alternatives, choosing the alternative that scores highest in this cue. With LEX, even minimal differences between two alternatives in the
inspected cue, may determine the outcome of the selection process. Lexicographic semiorder (LEX-Semi), on the other hand, is a LEX variant most suited to decision tasks with two alternatives, which allows for “just narrow differences” between them [26]. Namely, the cue parsing by the heuristic stops upon the first cue, where the difference between the two alternatives exceeds a predefined threshold.

The DEBA heuristic: The counterpart of LEX(-Semi) for binary cues is the Deterministic Elimination by Aspects (DEBA) heuristic in [19]. DEBA also ranks cues in order of decreasing importance \((x_1, x_2, \ldots)\) so that the alternatives are codified as ordered sequences of ones and zeros. It then inspects the value of all alternatives on \(x_1\) and eliminates all alternatives featuring a zero on it. The process is repeated when parsing the second and the remaining attributes until a single alternative remains. If more than one alternatives are left after all cues are inspected, or if there is a cue eliminating all remaining alternatives, a choice is made randomly among the currently surviving ones.

DEBA can also work with alternatives featuring continuous cues as long as these are “digitized” according to some rule before being processed by the heuristic. The simplest and most used such rule is called median split. For each cue \(x_i\), the rule computes the median \(m_i\) of the cue values across all alternatives. In alternatives, where the cue value exceeds \(m_i\), the cue is codified as 1(0) for favorable (adverse) cues and vice-versa. Hence, the median value \(m_i\) implicitly quantifies the satisfying threshold for cue \(x_i\): alternatives that score higher (or lower, if \(x_i\) is a negative one) than \(m_i\) are considered acceptable with respect to \(x_i\). Median split generates one binary cue out of each continuous one by dichotomizing it. More elaborate methods generate multiple binary cues out of a single continuous one. In [34], for instance, the full range of the cue values is split into multiple (rather than two) intervals and a binary attribute is assigned to each one of those. Then for each alternative only one of these cues can be non-zero.

Fast and frugal trees: Fast and frugal trees (FFTs) have been originally proposed with classification tasks in mind as binary trees [27], i.e., trees constructed with binary cues and a binary criterion, but can be generalized to other cases. They are decision trees with at least one exit leaf at every level. In other words, for every checked cue, at least one of its outcomes can lead to a decision.

As with DEBA, the notational convention is that cue profiles can be expressed as vectors of 0s and 1s. An ace corresponds to cue values which correlate more closely with “positive” outcomes (for example, presence of a disease in diagnostic tests). Convention also demands that left branches be labeled with 1s and right branches with 0s (ref. Fig. 2). Thus, each branch of the fully specified tree can be labeled according to the cue value associated with the node at the end of the branch.

The match of the lexicographic heuristics with the problem setting in II is straightforward. There is a one-to-one correspondence between choices and mobile crowdsensing tasks as well as between the attribute set \(a\) and \((x_1, x_2, \ldots)\). Their main advantage is that they can account for multiple determinants in the decisions of the contributors, as more often than not the case is with crowdsensing tasks.

2) Decision making under risk/uncertainty: The uncertainty that users may have about how other users will react (and consequently affect their payoffs) introduces a probabilistic element to the decision-making process (see the discussion in Section II). Formally, each alternative \(X\) may be described as a risky gamble with a minimum of four attributes

\[
X = (x_{min}, p; x_{max}, 1-p)
\]

Namely, choice \(X\) will generate a gain(loss) \(x_{min}\) with probability \(p\) and a gain(loss) \(x_{max}\) with probability \(1-p\).

The priority heuristic: The priority heuristic [6] is a lexicographic heuristic that accounts for choices with probabilistic rewards, i.e., risky gambles. Let us say we want to choose one of two alternatives \(X = (x_{min}, p; x_{max}, 1-p)\) and \(Y = (y_{min}, q; y_{max}, 1-q)\), no other attributes are inspected and the gamble with the higher value on its minimum outcome is chosen. Otherwise, if \(|p-q| > \theta^2\), a decision is made in favor of the gamble with the lowest probability of the minimum outcome without considering attributes beyond the second. In the opposite case, the gamble with the higher maximum outcome is chosen (and if the maximum outcomes are equal, a choice is made randomly). It has been shown analytically in [23] that the heuristic predicts a host of major empirical phenomena in risky choice such as violations of the common consequence and common ratio axioms as well as the four-fold pattern of risk attitudes [21].

The priority heuristic becomes relevant in MCS tasks, which induce competition between data contributors and condition their reward on the number of them who will contribute to the task (ref. to the relevant discussion in Section II). For example, when the task provider rewards only the first \(k\) contributors to the task with \(\pi^r\), the parameters of the heuristic could be taken to be:

\[
x_{max} = \pi^r \quad x_{min} = 0
\]

and \(p\) equals the probability, as inferred by the contributor, that she will be among the first \(k\) to serve the task. This probability is a function of \(\bar{n}\), the worst-case estimate the user can have about the competition for the task when the application provides information about the number of its subscribers in her proximity.

B. Strategic decision-making

1) Static models of strategic decision making: For some mobile crowdsensing tasks, a contributor’s payment may be a function of the choices made by other users. The normative solution for such games is the Nash Equilibrium (NE), or some refinement thereof if multiple equilibria exist. As with iDM,

---

1 Consider, as an example of a decision-making setting, the selection of a flat to rent. A human that prefers quiet suburban places, assesses the distance of a flat from the city business district positively: the further the better. Hence, the cue “distance” will be codified as 1 for flats located further than the median distance across all flats and 0 for the rest. The terms positive and negative cue are also relevant.

2 The value 0.1 is recommended for both thresholds \(\theta_1\) and \(\theta_2\) in [6].
vast experimental evidence of repeated deviations from this normative solution has motivated the investigation of other boundedly rational decision rules.

The Quantal Response Equilibrium [28] is a generalization of the NE that incorporates noisy, or stochastic, best-response at the equilibrium. The underlying idea is that “individuals are more likely to select better choices than worse choices, but do not necessarily succeed in selecting the very best choice”. The noisy/stochastic component reflects the random effects of estimation/computational errors, individual’s mood, or perceptual variations.

Other models of bounded rational strategic behavior relax the common knowledge assumption of rationality. Two of the most prominent examples are Level-$k$ theory [10] and the related Cognitive-Hierarchy theory [8]. In Level-$k$ theory, players are assumed to best respond to their beliefs about their opponent’s type; however, their beliefs are not constrained to be consistent or in equilibrium. A level-0 player is an unsophisticated player, usually assumed to simply randomize over his choices. A level-$k$ ($k > 0$) player assumes that an opponent is a level-$(k - 1)$ player and best responds to this belief. The level-$k$ model has been very successful at explaining many deviations from the NE for a wide array of games [11]. Empirical estimates of the distribution of types in the population predominantly find a larger proportion of L-1 and L-2 types and a smaller proportion of Nash or sophisticated types. This has important implications for the performance of commonly used approaches that assume that users are NE players. Examples of other heuristics applicable to sDM include choosing the action that: a) includes the maximum own payoff; b) maximizes the minimum payoff; c) maximizes social payoffs; d) minimizes the inequality in payoffs between players.

2) Learning and dynamic models of strategic decision making: The above models are predominantly used for static decision making, or one-shot games (although they can be extended in some cases to include elements of learning). The behavioral game theory literature has also tackled learning in repeated games, where a player may be matched with the same opponent for a number of rounds, or randomly rematched within a fixed population of players.

Learning models can be distinguished into belief-based, e.g., [9] and reinforcement learning models e.g., [13] by the type of variable that is updated on the basis of accumulated experience. In belief-based models, players form beliefs about their opponent’s behavior or type and update these beliefs as they gather more experience and feedback. For example, in the context of a level-$k$ model, players may have initial priors over the distribution of types in the population that can be updated as they interact with the population. In reinforcement learning models, players learn to associate payoff values with actions and update their estimates of the expected payoffs that are related to playing an action. Despite their apparent incompatibility, both belief-based and reinforcement learning models can be nested in a single model, the Experience Weighted Attraction model [7].

IV. USER PROFILING: TRAINING COGNITIVE HEURISTICS

A. Structural vs. non-structural models

In general, two classes of models can be used for user profiling purposes. The decision heuristics described in Section III represent structural models. They specify an explicit theory for the decision-making process and the relationship between the variables, namely the exact parametric form and the individual parameters. For example, in the case of DEBA, one would need to specify the way each data contributor ranks the task attributes in $a$ when selecting MCS tasks.

This is considerably different from current standard practices that heavily draw on non-structural models. Such models are agnostic about the parametric form of the relationship between the variables of interest. Non-parametric estimation techniques, such as the majority of machine learning approaches to modelling behaviour (e.g., neural networks or kernel density estimation) may instead try to learn this relationship. Alternatively, simple reduced-form modelling techniques such as a standard linear regression of the assumed explanatory variables on the dependent variable may be used. A key difference between the two approaches is that non-structural models seek to predict behaviour without giving any insight into the actual decision-making mechanism.

Non-structural models are inherently more flexible than their structural counterparts but this renders them more susceptible to overfitting. Consequently, such models may experience difficulty in generalizing well to new tasks for which no data has yet been collected [29].

B. The empirical dataset

The data we use hereafter to analyze decision-making heuristics have been collected through an online customized questionnaire. Fifty people, mostly graduate students of the

<table>
<thead>
<tr>
<th>Heuristic model name</th>
<th>Decision task type</th>
<th>Relevant task attributes</th>
<th>Aspects of bounded rationality</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEX [14], LEX-Semi [26]</td>
<td>iDM</td>
<td>any</td>
<td>limited information use</td>
</tr>
<tr>
<td>DEBA [19]</td>
<td>iDM</td>
<td>any</td>
<td>limited information use</td>
</tr>
<tr>
<td>Fast and frugal trees [27]</td>
<td>iDM</td>
<td>any</td>
<td>limited information use</td>
</tr>
<tr>
<td>Priority heuristic [6]</td>
<td>iDM</td>
<td>min/max payoffs, min payoff prob</td>
<td>limited information use</td>
</tr>
<tr>
<td>Level-k [10]</td>
<td>sDM</td>
<td>payoffs and beliefs</td>
<td>limited strategic sophistication</td>
</tr>
<tr>
<td>Cognitive Hierarchy [8]</td>
<td>sDM</td>
<td>payoffs and beliefs</td>
<td>errors/noisy choice</td>
</tr>
<tr>
<td>QRE [28]</td>
<td>sDM</td>
<td>payoffs</td>
<td>belief and reward updating from experience</td>
</tr>
<tr>
<td>EWA [7]</td>
<td>sDM</td>
<td>payoffs</td>
<td></td>
</tr>
</tbody>
</table>
Department of Informatics in AUEB, replied to the questionnaire from June 25th till July 3rd, 2015. The questionnaire invites its participants to consider that they are visiting the city center in their leisure time and receive offers about crowdsensing tasks on their smartphones. Each offer presents the user with two tasks (decision pair) and asks the participant to determine which one would (s)he choose to carry out. For each of the two tasks, two pieces of information are provided: the monetary payment $\pi$ awarded to those who carry out the task and the physical distance $d$ that a user would need to travel in order to get to the physical location of the task. Therefore, each offer constitutes an instance of a two-attribute choice problem with two alternatives.

The 50 questionnaire participants made 20 such choices sequentially, without having the option to go back to a previous choice and change it. The tasks that were paired within each offer were chosen carefully so that they present a tradeoff between the two attributes, reward and physical distance; that is, there was no instance where one task dominated the other by simultaneously featuring higher reward and smaller distance. The questionnaire can be retrieved online at http://5.101.107.163:8080/Questionnaire/ (in Greek).

### C. Training the models

Common to all lexicographic heuristic models is the need to determine the order in which cues are inspected by the heuristic. Let $X_i = (\pi_{xi}, d_{xi})$ and $Y_i = (\pi_{yi}, d_{yi})$, the pairs of alternatives presented to each questionnaire’s participant (decision pairs) and $S_i$, her choice in each of those, $i \in 1, \ldots, 20$. Then the order of cues can be determined by their ecological validities, $v_{\pi}$ and $v_{d}$. These express the predictive capacities of the cues and can be computed as

$$v_{\pi} = \frac{C_{\pi}}{C_{\pi} + I_{\pi}} \quad \text{and} \quad v_{d} = \frac{C_{d}}{C_{d} + I_{d}} \quad (5)$$

where $C_{\pi}(C_{d})$ are the number of correct inferences and $I_{\pi}(I_{d})$ the number of incorrect inferences based on the task reward(distance) cue alone, in cases that the corresponding cue discriminates between the two alternatives. The actual computations involved in (5) depend on the specific heuristic.

**LEX:** With LEX both cues are continuous. A cue will fail to discriminate only if it takes exactly the same value for both alternatives of a single decision pair (this is not the case with any of the 20 pairs in the questionnaire). Hence, for the task reward cue $\pi$ we could formally write,

$$C_{\pi}^{LEX} = \sum_{i=1}^{20} I(\{(X_i \succ Y_i \cap (\pi_{xi} > \pi_{yi})) \cup (X_i \prec Y_i \cap (\pi_{xi} < \pi_{yi}))\})$$

$$I_{\pi}^{LEX} = \sum_{i=1}^{20} I(\{(X_i \succ Y_i \cap (\pi_{xi} < \pi_{yi})) \cup (X_i \prec Y_i \cap (\pi_{xi} > \pi_{yi}))\})$$

and analogously for the task distance cue $d$, where $A \succ B$ denotes that option $A$ was selected over $B$ and $I(\cdot)$ is the indicator function, with $I(e) = 1$ if event $e$ is true.

**DEBA:** With DEBA, the continuous cues are “digitized” (turned to binary ones) via the median split method (ref. Section III-A1). Therefore, decision pairs may emerge, where one cue or both of them cannot discriminate. The expressions for $C_{\pi}$ and $I_{\pi}$ in the case of DEBA are given by (6), where $m_{\pi}$ is the median value of task rewards over all forty MCS tasks each questionnaire participant has been presented with.

The median values for the reward and distance cues were computed to be $m_{\pi} = 2.75$ and $m_{d} = 816.5m$, respectively.

Finally, fast and frugal decision tree(s) are constructed with the differences of task rewards and distances as cues. The $C_{\pi}$ and $I_{\pi}$ terms are given by (7), where $m_{\Delta \pi}$ denotes the median

---

**Fig. 1.** Histogram of the reward (top row) and distance (bottom row) cue validity values across all 50 participants for the three lexicographic models.
of the reward differences between the tasks involved in every decision pair.

Figure 1 provides ample evidence of the high heterogeneity in user preferences. Overall, approximately 60% of users appear to prioritize the physical distance of the MCS tasks over the reward given for carrying them out (30 vs. 19 for the LEX and DEBA models, and 29 vs. 21 for the fast and frugal tree considering the respective cue differences). However, they do so with highly variable intensity. Part of them (less than one quarter of the users for LEX and FFT, close to the half for DEBA) appear to rely almost exclusively on a single cue for making their choices, whether this is the reward or the physical distance of the task; whereas the majority of them alternate between the two cues.

D. Model comparison

We assess how well these lexicographic models can capture the choices of the participants following the standard two-step methodology for model evaluation. First, we check how well the model choices fit the choices made by the questionnaire participants. Hence, we use the full set of selections made by each participant as a training set for the models and then revisit each decision pair letting the models guide the selection of tasks. When averaging over all participants, LEX reports the highest score, closely followed by the fast and frugal tree model, as shown in Table IV-D.

3 One user appears to assign exactly the same importance to both attributes, i.e., \( \nu_x = \nu_d = 0.5 \).

Then we ask how well do the models generalize, i.e., predict choices in task selection instances other than those used for their training. Since a second independent dataset is not available, we resort to the cross-validation technique, in particular \( k \)-fold cross-validation. We partition the 20 decision pairs presented to each participant into \( k \) subsamples, out of which \( k-1 \) are used for the model training task and one is kept for testing the model. We let \( k \) values in \{2, 4, 5, 10, 20\} corresponding to test sample sizes of \{10, 5, 4, 2, 1\} decision pairs, respectively. The prediction capacity of the three models is plotted in Figure 3, and ranks them identically to the fitting scenario (ref. Table IV-D): first comes the LEX model, then the fast and frugal tree, whereas the DEBA model fares distinctly worse than the other two.

\[
C^D_{\omega} = \frac{1}{20} \sum_{i=1}^{20} \left( I\left( \{X_i \rightarrow Y_i \cap (\pi_{xi} > \pi_{yi})\} \cup \{X_i \leftarrow Y_i \cap (\pi_{xi} < \pi_{yi})\}\right) \right)
\]

\[
I^D_{\omega} = \frac{1}{20} \sum_{i=1}^{20} \left( I\left( \{X_i \rightarrow Y_i \cap (\pi_{xi} < \pi_{yi})\} \cup \{X_i \leftarrow Y_i \cap (\pi_{xi} > \pi_{yi})\}\right) \right)
\]

\[
C^{FFT}_{\omega} = \frac{1}{20} \sum_{i=1}^{20} \left( I\left( \{X_i \rightarrow Y_i \cap (\pi_{xi} - \pi_{yi} \geq m_{\Delta x})\} \cup \{X_i \leftarrow Y_i \cap (\pi_{yi} - \pi_{xi} \geq m_{\Delta x})\}\right) \right)
\]

\[
I^{FFT}_{\omega} = \frac{1}{20} \sum_{i=1}^{20} \left( I\left( \{X_i \rightarrow Y_i \cap (\pi_{xi} - \pi_{yi} < m_{\Delta x})\} \cup \{X_i \leftarrow Y_i \cap (\pi_{yi} - \pi_{xi} < m_{\Delta x})\}\right) \right)
\]

Fig. 3. Prediction accuracy of the three models when applying \( k \)-fold cross-validation to the training dataset (questionnaire replies), \( k \in \{2, 4, 5, 10, 20\} \).
TABLE II
FITTING THE QUESTIONNAIRE DATA.

<table>
<thead>
<tr>
<th>Heuristic model name</th>
<th>LEX DEBA</th>
<th>Fast and frugal tree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fitting score (%)</td>
<td>75.7</td>
<td>63.1</td>
</tr>
</tbody>
</table>

V. CONCLUSIONS

Our paper has attempted to challenge a couple of assumptions and common practices regarding the way the mobile crowdsensing task selection problem has been approached. At the same time, we tried to draw links between this problem and important modeling work in the area of cognitive sciences, which, to the best of our understanding, remains largely unknown to and unexploited by most of the (wireless) networking community. Although this work appears to have broader implications about several networking applications, our emphasis has been on the mobile crowdsensing paradigm. At least two research threads are naturally motivated by this work. The first one relates to accumulating further evidence that such cognitive heuristics are indeed being activated by end users when making choices of crowdsensing tasks. In the paper, we showed that some of these heuristics exhibit high predictive capacity and their activation could explain the experienced choices of end users. However, the empirical dataset (the questionnaire) is both small and artificial. The collection of real data about users’ choices, as realized through real crowdsensing applications running on their smartphones, would be necessary for a solid validation of the relevance of these heuristics to the MCS task selection process.

The second thread consists in exploring the implications of these heuristics for the performance and engineering of the crowdsensing platforms and applications. Practically, it treats questions such as “What is the expected performance of a crowdsensing system/application with users that think and act as agents of bounded rationality?” and “What kinds of incentives are due to motivate user contributions to specific tasks?”.

REFERENCES