Future Generation Computer Systems 87 (2018) 159–170

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Agreement-based credibility assessment and task replication in human computation systems

Lesandro Ponciano [a,b,*], Francisco Brasileiro [a]

[a] Federal University of Campina Grande, Bairro Universitário, Campina Grande, Paraíba, CEP 58429-900, Brazil
[b] Pontifical Catholic University of Minas Gerais, Bairro Coração Eucarístico, Belo Horizonte, Minas Gerais, CEP 30535-901, Brazil

HIGHLIGHTS

- Human computation is analysed from the perspective of task replication.
- An adaptive credibility-based task replication algorithm is proposed.
- Four metrics of credibility of participants are proposed.
- Adaptive algorithm reaches accuracy similar to non-adaptive one, using fewer replicas
- Difficulty of tasks affects participants' credibility and algorithm performance.

ARTICLE INFO

Article history:
Received 16 March 2017
Received in revised form 18 February 2018
Accepted 12 May 2018

Keywords:
Human computation
Credibility
Task replication
Inter-rater agreement

ABSTRACT

Human computation systems harness the cognitive power of a crowd of humans to solve computational tasks for which there are so far no satisfactory fully automated solutions. To obtain quality in the results, the system usually puts into practice a task replication strategy, i.e. the same task is executed multiple times by different humans. In this study we investigate how to improve task replication considering information about the credibility score of participants. We focus on how to automatically measure the credibility of participants while they execute tasks in the system, and how such credibility assessment can be used to define, at execution time, the suitable degree of replication for each task. Based on a conceptual framework, we propose (i) four alternative metrics to measure the credibility of participants according to the degree of agreement among them; and (ii) an adaptive credibility-based task replication algorithm that defines, at execution time, the degree of replication for each task. We evaluate the proposed algorithm in a diversity of configurations using data of thousands of tasks and hundreds of participants collected from two real human computation projects. Results show that the algorithm is effective in optimising the degree of replication, without compromising the accuracy of the obtained answers. In doing so, it improves the ability of the system to properly use the cognitive power provided by participants.

1. Introduction

Human computation is an emerging computing approach that draws upon human cognitive abilities to solve computational tasks for which there are still no satisfactory fully automated solutions [1–3]. Systems based on human computation are distributed systems that harness the cognitive power of a crowd of humans connected to the Internet to execute relatively simple tasks, whose solutions, once grouped, solve a problem that distributed systems equipped with only machines cannot solve satisfactorily. Such type of system has been proved to be effective in solving tasks that rely on human cognition such as detecting information in images [4], and processing natural language content [5], as well as more subjective tasks related to human's opinions and preferences [6].

There are currently two main types of human computation systems: *online labour markets* and *crowdsourced citizen science projects*. Online labour markets gather a crowd of humans willing to perform tasks in exchange for a relatively low financial incentive [7–9] — e.g. Amazon Mechanical Turk (mturk.com) and CrowdFlower (crowdflower.com). Crowdsourced citizen science projects, in turn, consist in a partnership between scientists and a crowd of humans willing to contribute to a scientific research, without receiving any financial incentive [10–12]. People acting in a citizen science project may contribute in a number of activities, which include performing human computation tasks. Examples of

* Corresponding author at: Pontifical Catholic University of Minas Gerais, Bairro Coração Eucarístico, Belo Horizonte, Minas Gerais, CEP 30535-901, Brazil.
E-mail addresses: lesandrop@pucminas.br (L. Ponciano), fubica@dsc.ufcg.edu.br (F. Brasileiro).

https://doi.org/10.1016/j.future.2018.05.028
0167-739X/

citizen science projects based on human computation are Stardust@home (stardustathome.ssl.berkeley.edu), in which people search for tiny interstellar dust impacts in images, and Galaxy Zoo (galaxyzoo.org), in which people perform morphological classification of galaxies from images.

To ensure quality in the execution of tasks, human computation systems usually require that the same task is executed multiple times by different humans; then, different aggregation mechanisms, which leverage the diversity and redundancy of multiple answers, can be employed to generate a more reliable answer to the task. In many systems, a task replication strategy is used as a way to obtain redundancy of answers in order to identify consensus in the set of answers or to tolerate faults that may cause some humans to generate wrong answers [11,13,3,14]. The *degree of replication* is the number of different humans who are performing each task. It is usually defined by the users at the moment of submitting a group of related tasks, all of them having the same degree of replication. Defining the suitable degree of replication for a task is a challenging process because it generates a trade-off between quality and cost. If the degree of replication is overestimated, an excessive amount of humans is used and, therefore, there is an increase in the cost of executing all tasks, perceived either financially or as a waste of resources that could have been allocated to do something else. On the other hand, if the degree of replication is underestimated, the desired quality in each answer is not achieved. Because tasks may differ among themselves in several ways, including its difficulty, it is expected that the ideal degree of replication can be different from one task to another, even when the description of the tasks are very similar. Given that users typically submit groups of hundreds or thousands of tasks, it is prohibitive to define manually a replication degree for each task addressing the cost–benefit trade-off. This is a typical situation in large citizen science projects based on human computation, as those hosted at the Zooniverse (zooniverse.org) and the Crowdcrafting (crowdcrafting.org) platforms.

This study analyses *how to automatically improve task replication at execution time by considering participants' credibility scores and the difficulty of tasks*. It focuses on (i) how to automatically measure the difficulty of tasks and the credibility of participants while they execute the tasks in the system, and (ii) how such measures can be used to define, at execution time, the suitable degree of replication for each task. To this end, we go through existing studies on human computation, credibility assessment, and task replication. Based on them, we propose four alternative metrics to measure participants' credibility considering the agreement among themselves. These metrics cover a diversity of participants' features, such as: the amount of generated answers; the amount of agreement with other participants that could be expected to occur through chance alone; and groups of participants that usually generate the most frequent answers. Then, we propose an adaptive task replication algorithm that optimises the degree of replication for each task, taking into account the participants' credibility and the difficulty of the task. The main idea is to stop replication as soon as the algorithm obtains a group of answers that is credible enough. Naturally, there are tasks in which the divergence in the answers is so high that a credible answer is not obtained even by increasing replication. The algorithm is designed to detect these situations, and stop replicating the task when a maximum degree of replication is reached.

Our evaluation study is based on trace-driven simulations [15]. The simulations are guided by data sets collected from two real human computation projects: Sentiment Analysis, and Fact Evaluation. Such data sets comprise hundreds of participants performing thousands of tasks, being valuable sources to analyse the performance of the proposed replication algorithm. In the simulations we evaluate 160 different configurations of the proposed algorithm and also two comparative strategies: (i) an *oracle* that knows whether an answer provided by a human is correct or not, and stops replicating the task when a correct answer is obtained; and (ii) a *majority voting* strategy that collects answers from a fixed number of participants, and identifies as correct the answer provided by the majority of them. We evaluate both the accuracy of answers and the replication reduction reached by these strategies.

The results show that the proposed credibility-based task replication algorithm is effective in achieving replication reduction while meeting other quality of service requirements, such as the required credibility. Some configurations of the algorithm reach higher accuracy than majority voting and achieves a replication reduction comparable with that attained by the oracle. In doing so, it improves the ability of the system to properly use the cognitive power provided by participants, while allows users to address the trade-off between different quality-of-service requirements.

The main contributions of this study are:

- we integrate concepts from four distinct literatures, which are human computation, credibility assessment, inter-rater agreement, and replication of tasks;
- we propose four alternative metrics to automatically measure the credibility of participants while they execute human computation tasks in a system, which are: surface agreement, experienced agreement, weighted agreement, and reputed agreement;
- we propose an adaptive task replication algorithm that optimises the degree of replication of each task according to participants' credibility, task difficulty and quality of service requirements.

These contributions have implications for human computation and related areas that are based on performing tasks with the participation of people, such as the areas of citizen science, crowdsourcing, and social computing. They also have implications for the area of distributed systems. Human computation systems are distributed systems in which computational resources are human beings. As such, some of the concepts employed in traditional distributed systems — i.e. those in which computational resources are machines — to replicate tasks can also be employed to replicate tasks in human computation systems. The study highlights this point, but also puts into perspective new issues in task replication that arise only in human computation systems.

The remainder of this paper is organised as follows. Firstly, we provide background on human computation, credibility assessment, task replication, and also discuss relevant previous work. Next, we present our approach to use agreement-based metrics to assess credibility and replicate tasks in human computation systems. Finally, we evaluate the proposed approach using data from two human computation projects, and then discuss the implications and limitations of the study.

2. Background and related work

Now we turn to present the terminology we adopt throughout the paper by briefly reviewing relevant notions of human computation, credibility assessment, and task replication. Thereafter, we discuss the related work.

2.1. Background

Human computation. Systems based on human computation are distributed systems in which humans participate as computational elements [16,2,3,17]. There are three core entities in this sort of system: requesters, workers, and platforms. *Requesters* are users who act in the system by submitting human computation tasks to be performed. A human computation task (or human intelligence

task, HIT) consists of some input data and a set of instructions about what to do with the data to produce a solution for the task. A human computation application comprises a group of tasks; each of them can be performed by one human. If the application is composed of a group of independent tasks, it is called a project or a bag of tasks. Otherwise, if composed of a group of tasks organised in a sequence of connected steps, it is called a workflow. *Workers*[1] are participants who act in the system as human computers by executing the human computation tasks. The solution provided by a worker to a task is called an *answer*.[2] Workers usually perform tasks independently, i.e. they do not know which tasks are being performed by others and which answers are being provided. Finally, *platforms* are systems that act as a middleware receiving requesters' tasks and managing their execution by the workers.

Human computation tasks can be classified according to their granularity and subjectivity. In terms of granularity, tasks can be broadly classified into *micro-tasks* and *macro-tasks*. Micro-tasks consist of few instructions that require little time to perform, such as a few minutes. Macro-tasks, in turn, consist of several instructions that require a long time to perform, such as hours or days. Tasks can also be classified into *factual* or *non-factual* according to the degree of subjectivity of their instructions. Tasks are non-factual when their instructions contain many aspects of subjectivity, such as opinion, feeling and creativity. For example, a task that displays two sunset images and asks the worker to choose the image that depicts the most beautiful sunset is defined as non-factual. The answers generated by workers in this kind of task are defined neither correct nor incorrect. On the other hand, in factual tasks, the instructions are more precise and the answers can be evaluated in terms of correctness by a human expert. For example, a task that displays an image of a landscape and asks workers whether there is a tree in the landscape depicted in the image is defined as factual.

Human computation tasks can also differ among themselves by complexity and difficulty. Although there are similarities between these concepts, they are neither independent nor equivalent [19]. *Complexity* can be defined as the amount of cognitive effort the task demands. *Difficulty*, in turn, is an attribute of both task and human (or group of humans) who is (are) performing the task. It includes human factors such as familiarity with the instructions, amount of knowledge and past experience. Thus, while the complexity of a task is similar to all humans who perform it, the perception of difficulty may vary from one human to another. Moreover, the complexity of a task is constant, while the perceived difficulty of a task by a particular worker can decrease or even increase when evaluated at different time instants [20].

Credibility assessment. Conceptual frameworks have been proposed to define the concept of credibility, to understand the elements related to this concept in computational systems, and to support its study in a multidisciplinary perspective [21–23]. *Credibility* means "believability" and it is mainly associated with the notions of trustworthiness and expertise [21]. Credibility is also associated to the concepts of reputation and relevance [22]. From this point of view, a high credible person in a specific domain of knowledge is someone known for having high expertise in such domain, and being able to be trusted. So, users can rely on what such person says. This concept may be helpful to identify good workers in human computation platforms. The components of a system that are subject to credibility assessment are the pieces of information referred to as *messages*, the *sources* that produces such messages, and the *medium* in which the messages are transmitted or delivered [23]. These concepts can be used to analyse credibility in human computation systems. In this sort of system, the messages are the answers to the tasks, the sources are the workers who produce such answers, and the medium is the platform in which the tasks are performed, and that delivers answers to the requesters.

The assessment of credibility consists of prominence and interpretation [23]. *Prominence* focuses on making relevant elements noticeable. If an element is not noticed, it will not have an impact on the credibility. *Interpretation*, in turn, is the judgement about each noticed element. The interpretation of each element determines how it impacts the credibility of the subject under evaluation. Four types of assessments can contribute to the prominence: presumed, reputed, surface, and experienced [21,23]. *Presumed* assessment is based on general assumptions that the evaluator has on his/her mind about the subject under evaluation. *Reputed* assessment is based on what third parties have reported about the subject under evaluation. *Surface* assessment defines the credibility of the subject based only on a simple inspection of some of its elements. Finally, *experienced* assessment is based on first-hand experience with the subject under evaluation.

We draw on these types of assessment to derive metrics to perform the prominence and interpretation of workers' credibility. The proposed credibility metrics are based on the level of agreement between workers when performing the tasks. The concept of "agreement among humans" has been covered in the literature on the subject of inter-rater agreement (or inter-rater reliability) [24–26]. It focuses on how much consensus there is in the answers given by humans. High consensus among the workers does not guarantee high accuracy in the answer to the task, but certainly there is no high accuracy if the agreement is low [25]. Considering this literature, we determine the credibility of a worker based on his/her level of agreement with other workers in each individual task.

Task replication. *Replication* is a mechanism used when it is important to obtain some kind of redundancy. It may consist of storing the same data on multiple storage devices or executing the same instructions many times. Such kind of redundancy is important in both human-based and machine-based systems. For instance, redundancy is a fundamental building block to achieve high availability/performance or to tolerate faults in distributed systems [27,28]. In social choice and inter-human agreement contexts, redundancy of answers from different humans is used to elicit preferences, opinions, and collective choices [24,25]. The following concepts coined in those areas are relevant to the present work: active replication, passive replication, degree of replication, and aggregation.

Active and passive replication are concepts used in distributed systems [27]. In *active replication*, each replica of a task is fully executed by different computing elements, with each non-faulty element starting from the same initial state and getting to the same final state of the task. In *passive replication*, while the task is executed in one primary computing element, a backup of the state of the execution is maintained in other secondary computing elements. This backup keeps the computations already performed and can be restored if a failure occurs during execution. In human computation, the concept of passive replication is possible in macro-tasks in which workers generate a sequence of answers to the same task. In such cases may be important to keep copies of partial answers. The present study focuses on micro-tasks and on active replication. Thus, the same task is fully executed multiple times by different workers in order to obtain redundancy of answers. Such redundancy is used to identify consensus in the set of answers and to tolerate faults that may cause some human failures, e.g. lapses, slips, and mistakes [20].

[1] Workers are called *providers*, crowdworkers or *turkers* in studies focused on online labour markets [2,3]. They are also called *volunteers* or *citizen scientists* in studies on crowdsourced citizen science and scientific crowdsourcing [18].

[2] Answers are also referred in the literature as results, outcomes, outputs, or responses.

As we discussed earlier, the degree of replication is the number of times the task is replicated to different workers. Identifying the appropriate degree of replication is a challenging process because of its inherent cost–benefit trade-off. When different replicas of a task produce different answers, a strategy of *aggregation of answers* is used to identify the final answer to the task. Thus, the strategy of aggregation defines which answer should be used. An example of strategy is to consider correct the most frequent answer. The degree of replication can have effects on the quality of the answer obtained in the aggregation. The present study investigates the role that credibility metrics play in both defining the degree of replication and performing the aggregation of answers in human computation tasks.

2.2. Related work

To the best of our knowledge, no previous studies have examined the credibility assessment and task replication framework in the context of human computation systems. However, we can identify some related work in the context of quality assurance, aggregation of answers, and task redundancy.

Quality assurance. Workers may not properly perform tasks for several reasons [29], such as: (i) cheating behaviour, (ii) lack of ability and expertise, and (iii) problems in the definition of the task. A *cheating* behaviour occurs, for example, when workers put little effort to perform the task, or provide wrong answers to tasks intentionally. It may occur in online labour markets in which some workers are only interested in receiving the money regardless whether the task is being properly performed or not [2,30]. Workers can also actively react to actions from the requester, such as planning collusion against requesters who have submitted poorly designed tasks [31]. When analysing the answers received from the system, it is important to be careful in judging the workers; they usually complain about being unfairly labelled as bots, spammers, cheaters, etc. [32]. We highlight that the occurrence of cheating workers is found to be negligible in citizen science projects [11].

The main causes of errors in answers to human computation tasks are *problems on the task design* and *workers' lack of ability/expertise* [33,34]. Problems on the design of task, e.g. ambiguous instructions and cognitive overload, are usually addressed by following platform guidelines and conducting pilot tests. The way requesters deal with lack of ability/expertise of workers varies if the task is factual or non-factual. In factual tasks, the assessment of the expertise of workers is usually made by using pre-task qualification tests, gold standard data sets, and workers' behavioural measures [13,35,29]. In non-factual tasks, in turn, there is no unique answer, so workers are not evaluated in terms of expertise, therefore, those strategies do not apply. In this case, post-task quality assurance strategies based on redundancy and aggregation are widely used [25,29].

Aggregation of answers. Aggregation focuses only on getting accurate or relevant answers to the tasks. It has been used as an offline procedure, performed after all answers have been collected from the system [16,36,37]. Several strategies for aggregation of answers have been proposed in the last few years; a number of them are reviewed by previous studies [16,36,38]. Many strategies are based mainly on detecting agreement or consensus by using, for example, an expectation–maximisation (EM) algorithm. However, the simplest and most used strategy is majority voting [36]. Previous studies on output aggregation fall short in investigating how credibility metrics (inferred from the dynamics of workers in the system) can be considered by an aggregation strategy, and how they can help one to optimise the system effectiveness, not only in terms of accuracy of answers, but also in terms of performance requirements, such as the number of replicas, and the urgency to obtain a final answer. The present study investigates these issues and examines how the aggregation can be performed, online, as part of the task replication process.

Task redundancy. If the proportion of workers who generate wrong answers is known, one can define the redundancy per task before submitting the tasks to the system [39]. However, workers perception of difficulty, and their probability of providing wrong answers, can greatly vary, both with the characteristics of tasks (e.g. instruction and input data), as well as with workers' characteristics (e.g. familiarity and experience), making it hard to estimate, beforehand, the proportion of workers who provide wrong answers. Some studies have sought to identify the number of workers who must perform a task, so that accurate answers are obtained. For example, in the first release of the Galaxy Zoo project, each task was executed on average 38 times [11], and a recent study suggests that each task should be replicated 10 to 11 times in Amazon Mechanical Turk [14]. However, pre-establish an equal degree of replication for tasks that may have differences in terms of difficulty may not be effective. Tasks that are more difficult tend to lead workers to disagree more on the answers they provide [40,41], and require more replicas to reach a suitable level of accuracy. The present study proposes a strategy to adaptively define, at execution time, the degree of replication, according to estimates of the difficulty of tasks and workers' credibility.

Studies have highlighted several performance requirements that are important to requesters, such as time, cost, accuracy, reproducibility, and security [2,8,3]. By reducing the number of replicas that are generated, and taking into account workers' credibility, a replication algorithm may reduce costs and increase the accuracy of the answers obtained from the systems. However, it can also introduce some delays in the execution of tasks. For example, tasks can take longer to execute depending on whether their replicas are generated sequentially or in parallel. If the user has a time requirement, generating replicas sequentially may be a drawback. To cope with this issue, the replication algorithm proposed in this paper allows users to inform the level of urgency to get a final answer. It is designed so that the higher the urgency, the higher the parallelism in the execution, but the lower the chance to reduce the number of replicas used.

In general, these previous studies clarify several aspects of workers' behaviour, quality assurance, and aggregation of answers in human computation systems. However, little progress has been made in terms of understanding how to estimate the credibility of workers and to optimise the replication of tasks. This fact constitutes an important shortcoming because a key feature of this kind of system is its capacity to deal with the difference of performance among the workers and to optimise the use that the system makes of the cognitive power provided by them. The present study shows that agreement-based credibility metrics provide us with a clear understanding of workers proficiency and a credibility-aware task replication is effective in optimising the performance of these systems.

Our initial studies have analysed the potential of optimising the redundancy of tasks according to the worker's credibility [42], and mapping the concept of task replication onto the context of human computation system [43]. Besides integrating the notion of credibility assessment and task replication in human computation system, the present paper presents our approach to use credibility metrics to improve task replication. We propose (i) a set of four alternative metrics to measure the credibility of workers in human computation, and (ii) a task replication strategy that optimises the degree of replication according to workers' credibility score and taking into account requesters' requirements, such as required credibility, urgency and maximum number of replicas allowed.

3. Agreement-based credibility assessment and task replication

Our approach consists of three major components: (i) measuring the worker's credibility considering the difficulty of tasks; (ii)

measuring the credibility of each answer and groups of answers; and (iii) replicating tasks according to the value of credibility metrics. Now we turn to discuss each of these components.

3.1. Measuring the credibility of workers

We establish four alternative metrics for automatically measuring the credibility of workers. Each metric covers a different way of assessing credibility: surface, experienced, presumed and reputed. We consider that the credibility of workers varies with the degree of difficulty of the task. The degree of difficulty of each task is measured by using Shannon's entropy [44]. This entropy measures the degree of divergence among the answers to the task provided by distinct workers. The idea behind this metric is that the greater such divergence, the more difficult the task is. The difficulty degree is denoted by d and defined by Eq. (1). In this equation, S denotes the set of unique answers to the task, each $a \in S$ denotes one distinct answer, and $\Pr(a)$ denotes the proportion of workers who provided the answer a. When all the workers provide the same answer to the task, the difficulty degree assumes the minimum value ($d = 0$). The value of d is impacted by both the diversity of answers received to the task and the distribution of workers in these answers. In order to categorise the tasks according to their degree of difficulty, we enumerate the possible values of d by rounding them to one decimal place $d = 0.0, 0.1, 0.2, \ldots$. Using such categorisation, we estimate a value of worker's credibility for each degree of difficulty of tasks.

$$d = -\sum_{a \in S} \Big(\Pr(a) \times \log_2 \Pr(a) \Big). \tag{1}$$

Surface agreement. This credibility metric is the probability of a worker to provide an answer equal to the answer provided by the majority of workers who performed the task. Let $n_{i,d}$ be the total number of tasks with degree of difficulty d performed by a worker i, and let $f_{i,d}$ be the amount of those tasks in which the worker i provided the same answer provided by the majority of workers who performed the task. The joint probability of agreement of answers provided by the worker i and answers provided by the majority of workers is computed as defined by Eq. (2). When $s_{i,d} = 1$, there is a complete agreement between the answer provided by worker i and the answer provided by the majority in all the tasks performed by worker i. On the contrary, the closer $s_{i,d}$ gets to 0, the lower the agreement between worker i and the majority.

$$s_{i,d} = \frac{f_{i,d}}{n_{i,d}},\ s_{i,d} \in [0, 1]. \tag{2}$$

Experienced agreement. This credibility metric is based on Cohen's kappa statistic [24]. It has been used to measure the agreement between the answers provided by two people. Here, we use this metric to measure the degree of agreement between the answers provided by a worker and the answers provided by the majority of workers to the same tasks. Unlike surface credibility, experienced credibility takes into account not only the amount of joint probability of agreement ($s_{i,d}$), but also the amount of agreement that could be expected to occur through chance alone ($c_{i,d}$), i.e. the probability of workers to agree giving random answers to the tasks. The experienced agreement is denoted by $e_{i,d}$ and formalised in Eq. (3). To keep the values in the range between 0 and 1, as occurs in the other metrics, we use the value $(e_{i,d}+1)/2$. When the result is 1, there is a complete agreement between worker i and the majority. If it is higher than or equal to 0.5, the agreement is higher than or equal to the chance-expected agreement. Finally, a value lower than 0.5 indicates that the measured agreement is lower than the chance-expected agreement.

$$e_{i,d} = \frac{s_{i,d} - c_{i,d}}{1 - c_{i,d}},\ e_{i,d} \in [-1, 1]. \tag{3}$$

Weighted agreement. People usually assume that the more information is used to estimate the credibility, the more likely the estimation is accurate. For example, the estimation of credibility based on only one answer provided by a worker seems to be less reliable than the estimation of credibility based on 10 answers provided by that worker. We propose a presumed credibility metric that reflects this idea. It is denoted by $p_{i,d}$ and defined in Eq. (4). This metric is a weighted harmonic mean between the joint probability of agreement ($s_{i,d}$) and the neutral credibility of 0.5. The weight of $s_{i,d}$ is the number of tasks performed by the worker ($n_{i,d}$) and the weight of 0.5 is 1. It means that the larger the number of tasks performed by the worker ($n_{i,d}$), the greater the weight of the estimated probability of agreement in the credibility of the worker. Thus, when $n_{i,d} = 0$, the weighted agreement $p_{i,d}$ is 0.5, which indicates neutral credibility. In the proportion with which the number of tasks performed by the worker ($n_{i,d}$) increases, the value of weighted agreement of the worker tends to $s_{i,d}$.

$$p_{i,d} = \frac{n_{i,d} + 1}{\frac{n_{i,d}}{s_{i,d}} + \frac{1}{0.5}},\ p_{i,d} \in [0, 1]. \tag{4}$$

Reputed agreement. The idea behind this credibility metric is that the credibility of a worker should be increased when he/she agrees with highly credible workers, and should be decreased when he/she disagrees with highly credible workers. Thus, the purpose of our reputed credibility is to consider in the credibility of a worker i the scores of credibility of the other workers with whom he/she agreed and disagreed in the past. Let $Y_{i,d}$ be the set of workers with whom i agreed when he/she provided the majority answer. We define $y_{i,d}$ as the averaged credibility of this set of workers; it is computed by using Eq. (5). Let $X_{i,d}$ be the set of workers with whom i disagreed when he/she did not provide the majority answer. We define $x_{i,d}$ as the averaged credibility of this set of workers; it is computed by using Eq. (6). Once defined $y_{i,d}$ and $x_{i,d}$, we can now compute the reputed agreement of worker i by using Eq. (7).

$$y_{i,d} = \frac{\sum_{w \in Y_{i,d}} s_{w,d}}{|Y_{i,d}|} \tag{5}$$

$$x_{i,d} = \frac{\sum_{w \in X_{i,d}} s_{w,d}}{|X_{i,d}|} \tag{6}$$

$$r_{i,d} = \frac{s_{i,d} + y_{i,d} - x_{i,d} + 1}{3},\ r_{i,d} \in [0, 1]. \tag{7}$$

The credibility $r_{i,d}$ assumes the minimum value 0 when the following conditions are satisfied: (i) the worker i disagreed with the majority in all the tasks he/she performed (i.e. $s_{i,d} = 0$), (ii) as the worker i never agreed with the majority, there is no gain of credibility from other workers that provided the majority answer (i.e. $Y_{i,d}$ is an empty set and therefore $y_{i,d} = 0$), and (iii) the workers with whom the worker i disagreed and that provided the majority answer have the highest credibility score, so the worker i loses 1 in credibility for disagreement (i.e. $x_{i,d} = 1$). On the other hand, the credibility $r_{i,d}$ assumes the maximum value 1 when the following conditions are satisfied: (i) the worker i agreed with the majority in all the tasks he/she performed (i.e. $s_{i,d} = 1$), (ii) the workers with whom he/she agreed have the highest credibility score, so there is the maximum gain of credibility from these workers (i.e. $y_{i,d} = 1$), and (iii) as the worker i never disagreed with workers who provided the majority answer, there is no loss of credibility for disagree with such workers (i.e. $X_{i,d}$ is an empty set and $x_{i,d} = 0$).

These credibility metrics fit well into our objective of considering a diversity of credibility aspects of workers' behaviour. Surface agreement is the simplest metric; it takes into account only the raw agreement among workers. Experienced agreement,

in turn, measures the real agreement by deducting from the raw agreement the amount of agreement that may occur simply by chance. Weighted agreement weighs the effect of the amount of data used to compute the degree of agreement. Finally, reputed agreement takes into account not only the amount of agreement exhibited by a worker, but also the credibility of the workers with whom he/she agreed or disagreed.

3.2. Measuring the credibility of answers and groups of answers

The *credibility of an answer* is the credibility of the worker who generates it. Thus, for example, when a worker w with credibility 0.8 performs a given task, the answer provided by such worker to such task has credibility 0.8. Equal answers generated by different workers for the same task are matched together into groups of answers. For a task, we have g groups of answers, each denoted by G_a, for $1 \leq a \leq g$. The credibility $C(G_a)$ of a group of answers G_a is the probability of the answers in the group be good and the answers in the other groups for the same task be bad. It is computed according to the credibility of the answers in the group as expressed by Eq. (8).

$$C(G_a) = \frac{P(G_a \text{ good}) \prod_{i \neq a} P(G_i \text{ bad})}{\prod_{j=1}^{g} P(G_j \text{ bad}) + \sum_{j=1}^{g} P(G_j \text{ good}) \prod_{i \neq j} P(G_i \text{ bad})}. \quad (8)$$

In this equation, $P(G_a\text{good})$ is the probability of the results in the group G_a being good, computed as $\prod_{i=0}^{|G_a|} C(R_i)$ for all answers R_i in the group of answers G_a and where $C(R_i)$ is the credibility of the worker who provided the answer R_i. Correspondingly, $P(G_a\text{bad})$ is the probability of the answers in the group of answers G_a being bad, computed as $\prod_{i=0}^{|G_a|}(1 - C(R_i))$ for all answers R_i in G_a. This approach was proposed to identify groups of bad answers generated by machines in volunteer computing systems [45]. In this paper we use it to measure the credibility of a group of answers provided by a group of human workers when performing a human computation task of a given degree of difficulty. Our main goal is to identify the most credible group of answers. Formally, a group of answers G_a is the most credible if $C(G_a) > C(G_b)$, for $1 \leq b \leq g$, and $b \neq a$.

3.3. Replicating tasks according to credibility metrics

The main idea behind the credibility-based task replication is to use credibility metrics to define, at execution time, whether more replicas to the task are required. It is based on the three types of credibility metrics described in the previous sections: *credibility of workers*, *credibility of answers*, and *credibility of groups of answers*. Algorithm 1 shows the sequence of steps[3] computed in the replication of a human computation task.

Given a task t, and a credibility metric m, the goal of the Algorithm 1 is to return a final answer to the task and the credibility associated with that answer. This is done by attempting to generate a minimum number of replicas and considering the following restrictions:

- Minimal credibility in the final answer to the task (parameter *reqCred*). It is a decimal value between 0 and 1 which indicates the desired credibility level for the final answer obtained by the algorithm, so that it is considered credible by the requester.
- Maximum number of replicas (parameter *maxRepl*). It is a positive integer value greater than 0 which indicates the maximum number of replicas that can be generated by the algorithm.

[3] For simplicity and clarity of computing steps, the algorithm is presented without performance optimisation in the computations performed by digital computers.

ALGORITHM 1: Credibility-based Task Replication

input : Task t, Credibility metric m, Required credibility *reqCred*, Maximum number of replicas *maxRepl*, Urgency *urge*
output: Final answer to the task *finalAnswer*, Credibility of the final answer *finalCred*;

```
1  countRepl ← 0;        /* The total number of replicas already
   generated by the algorithm. */
2  S_t ← {};             /* Map of works who provides each answer. */
3  numReplPerTurn ← max(⌊maxRepl × urge⌋, 1);
4  repeat
5      numRepl ← min(numReplPerTurn, maxRepl − countRepl);
6      createReplicas(numRepl, t, S_t);   /* It creates numRepl
       replicas of task t, waits for their answers, and stores
       these answers and respective worker ids in the map S_t.
       */
7      G ← computeWorkersCredibility(S_t, m);   /* It computes the
       credibilities of workers using credibility metric m;
       the initial credibility of a worker is set to 0.51. */
8      finalAnswer, finalCred ← getTheMostCredibleGroupOfAnswer(G);
       /* It computes the credibilities of groups of answers
       using Equation 8. */
9      countRepl ← countRepl + numRepl;
10 until finalCred ≥ reqCred or countRepl = maxRepl;
11 return finalAnswer, finalCred;
```

- Urgency level of execution (parameter *urge*). It is a decimal value between 0 and 1 which indicates the level of urgency to obtain a final answer to the task. If the requester wants that all replicas of a task be generated at once, he/she sets the value of urge to 1. If the requester admits that replicas are sequentially generated one after another to conclude, he/she sets the value of urge to 0.

The number of replicas executing in parallel (variable *numReplPerTurn*, line 3 of Algorithm 1) varies according to the maximum number of replicas and the urgency. When the urgency assumes the value 1, all replicas (i.e. the maximum limit) are generated at once. In this case, the algorithm has no possibility to minimise the number of replicas that will be generated. On the other hand, when the urgency is 0, replicas are created sequentially. At each iteration of the loop repeat-until, the algorithm generates *numRepl* replicas of the task, waits for their answers, calculates workers' credibility, groups similar answers received from them, calculates the credibility of each group of answers, and identifies the most credible group of answers. A newcomer worker, which is yet to execute a task, has its initial credibility set to 0.51. The replication can stop either because the maximum number of replicas was reached, or because the most credible group of answers exhibits credibility equal or larger than the required credibility. The algorithm reduces the number of replicas used when it stops the replication before the maximum limit of replicas is reached. The minimum number of replicas that can be generated by the algorithm is 1. It occurs, for example, when the urgency is set to 0, and the required credibility is smaller than the credibility of the first worker that provides an answer.

The replication may end because the maximum number of replicas was reached, but without satisfying the credibility requirement. This occurs when an excessive amount of replicas has been used and convergence to a final answer has not occurred. In this case, two approaches are possible: conservative and non-conservative. In a *conservative perspective*, tasks that do not reach the required credibility threshold are marked as "inconclusive" and their answers are ignored. This is especially important when seeking the correct answer of factual tasks. Inconclusive tasks may have some features that the requester wants to further investigate, or should be resubmitted to be analysed by more skilled workers.

On the other hand, in a *non-conservative perspective*, all answers are used even when the required credibility is not reached. The non-conservative perspective is important in non-factual tasks, in which the concept of correct answer is absent, and requesters do not know a priori the level of agreement that can be expected. Note, however, that even in this case, it is possible to reduce the required replication for some tasks, by setting an appropriate value for the required credibility parameter.

The algorithm chooses the final answer to the tasks based on the credibility of the workers who provided the answers. Therefore, if a majority of workers provide the same incorrect answer, this answer is only taken as the final answer if the credibility of this group of answers is larger than the credibility of all other groups of answers provided by the other workers that answered the task. This may not be the case, if the majority group is formed by workers that are less credible than the workers in another minority group.

4. Evaluation

We evaluate the proposed approach by using trace-driven simulations [15]. Before presenting and discussing the results, we detail the data sets used to guide the simulations, the simulation model, and the evaluated scenarios.

4.1. Data sets

We use data collected from two human computation projects: Sentiment Analysis, and Fact Evaluation. Tasks in the Sentiment Analysis[4] project ask workers to judge the sentiment expressed in a tweet[5] about the weather condition. The possible answers to such tasks are: "negative", "neutral", "positive", "tweet not related to weather condition", and "I can not tell". The data set of this project consists of 569,375 replicas of 98,979 tasks that were performed by 1958 workers. Tasks in the Fact Evaluation[6] project, in turn, ask humans to judge facts about public figures on Wikipedia, an example of fact is "Stephen Hawking graduated from Oxford". The possible answers to this kind of tasks are: "yes, the fact is correct", "no, the fact is not correct", and an option to skip if the worker is unsure. The data set of this project consists of 220,000 replicas of 42,624 tasks that were performed by 57 workers. In both Sentiment Analysis and Fact Evaluation projects, a set of *ground truth tasks* — tasks in which the correct answers are known — is available. There are 300 ground truth tasks in the first project, and 576 in the other. The notable differences between the two projects turn them valuable to evaluate the performance of the algorithm in distinct scenarios.

4.2. The simulation of task replication and the computation of performance metrics

The traces that guide the simulation are the data sets discussed in the previous section. They provide the temporal order in which the tasks are performed, each answer generated by every worker, and the answers to the ground truth tasks. Simulations allow us to evaluate a large number of configurations of the proposed approach based on behaviour of workers in real systems.

Simulation of the credibility-based task replication algorithm. The dynamics of the trace-driven simulator are as follows. The task replication algorithm is called for each task in the trace, with *maxRepl* set to the number of replicas for the task available in the trace. The algorithm performs the task replication taking from the trace *numRepl* answers received to the task and the workers that provided such answers. Then, it calculates the credibility of the workers that provided these answers, and the credibility of the answers and group of answers. After that, it decides whether to generate another set of replicas for the task. This process is repeated until the halt condition is met (line 10 of Algorithm 1).

Simulation of the comparative strategies. Besides the proposed task replication algorithm, we also simulate two comparative strategies: majority voting and oracle. For the *majority voting* strategy the final answer to each task is that provided by the majority of workers who performed the task. We consider two cases: one that uses only 3 answers, i.e. the smallest level of replication possible for majority voting, and another that uses all answers in the trace. These configurations provide a trade-off between accuracy and replication cost. The *oracle*, in turn, knows whether a worker provides a correct answer or not. For each task stored in the trace, it reads the answer provided to the replicas sequentially. When it reads a correct answer, it stops replicating the task. Thus, the oracle stops replication as soon as the first correct answer is received from a worker. If none of the received answers is correct, the oracle uses all the answers stored in the trace and ends the replication with the last read answer.

Computation of performance metrics. In each simulation, as the trace is processed, the simulator generates the following outputs:

- Replication reduction. It is a measure of the proportional reduction in the number of used replicas. For example, if there are 5000 replicas in the trace and the proposed algorithm ends the replication generating 3000 replicas, the replication reduction is $(5000 - 3000)/5000 = 0.4$.
- Accuracy. It is the hit rate of final answers chosen by the algorithm and the answers in the ground-truth tasks. For example, if there are 300 ground truth tasks in the project and in 210 of them the algorithm reached an answer equal to the ground truth answer, the accuracy is $210/300 = 0.7$.
- Proportion of inconclusive tasks. It is the proportion of tasks for which the proposed algorithm does not reach a final answer that meets the level of credibility required by the requester. For example, if the requester submits a set of 1000 tasks and requires a level of credibility of 0.95, but the algorithm reached at least such credibility in only 900 tasks, the proportion of inconclusive tasks is $(1000-900)/1000 = 0.1$.

By construction, the accuracy reached for the oracle is the highest possible. It can be less than 1 in cases where there are tasks for which none of the answers received to the task is a correct answer. The oracle provides the best replication reduction that can be achieved without compromising accuracy. A strategy that interrupts replication before a correct answer is obtained, can achieve a higher replication reduction than the oracle, but with an associated cost in terms of reduced accuracy. The maximum number of replicas that the proposed algorithm and the oracle can generate to the task in the simulation is limited to the number of answers available in the trace. One of the main objectives of the evaluation is to verify to what extent the strategies are able to get an accurate answer to the task and interrupt its replication before using all the answers (replicas) available in the trace. The majority voting strategy is a very simple, and commonly used, aggregation method, and provides a kind of lower bound on the accuracy that can be achieved.

[4] Available at https://sites.google.com/site/crowdscale2013/shared-task/sentiment-analysis-judgment-data.

[5] Messages shared in the social network Twitter (http://twitter.com/).

[6] Available at http://googleresearch.blogspot.com.br/2013/04/50000-lessons-on-how-to-read-relation.html.

Table 1
Configurations of the task replication.

Independent variables	Credibility metric (surface, experienced, reputed, weighted) Required credibility (0.6, 0.7, 0.8, 0.91, 0.93, 0.95, 0.97, 0.99) Urgency (0, 0.25, 0.5, 0.75, 1)
Dependent variables	Replication reduction Accuracy Proportion of inconclusive tasks
Comparative scenarios	Lower bound (majority voting with 3 replicas, and with all possible replicas) Upper bound (oracle)

4.3. Configurations of the simulations

We simulate several configurations of the proposed algorithm (Table 1). Each configuration is a combination of the levels of the independent variables (credibility metric, required credibility and urgency). The effect of varying such levels was measured by using three dependent variables, that are the performance metrics defined above.

The combination of the levels of the independent variables in each data set results in a total of 160 different configurations of the proposed algorithm. To identify the best configurations, we used a multi-objective optimisation analysis based on the concept of Pareto front [46]. The main objectives are to maximise replication reduction and the accuracy of the answers. In the conservative perspective, an additional objective is to minimise the proportion of inconclusive tasks.

By using Pareto front analysis, we want to identify the set of dominant configurations in terms of their effects on the dependent variables. Dominant configurations are those that show better results than those presented by the dominated configurations. A configuration A dominates a configuration B if A outscores B in at least one objective, and A is not worse than B in any of the objectives.

Because some of the algorithms can stop the replication of a task before using all the answers available in the trace, the order in which the answers to the task appear in the trace can impact the effectiveness of the algorithms. This impact was measured by running 5 simulations of each configuration of the algorithms, using the answers to each task in the trace sorted randomly. In analysing the results, we always show the average of the results obtained in these simulations with error bars to a confidence level of 95%.

4.4. Results

We first present the results on workers' credibility and the difficulty of tasks. After that, we present the results about the performance of the task replication.

Difficulty of tasks and credibility metrics. The studied projects differ among themselves in terms of the degree of difficulty of their tasks. There are 24 degrees of difficulty in the Sentiment Analysis project ($min = 0$, $max = 2.3$) and 6 degrees of difficulty in the Fact Evaluation project ($min = 0$, $max = 2.7$). Using data from all workers and all tasks, we estimate the credibility of each worker in every possible degree of difficulty.

We can measure the difference between the values of credibility estimated by using two different credibility metrics as a mean absolute difference. Let x and y be two credibility metrics, the mean absolute difference for these metrics ($m(x, y)$) in a set of workers W is defined as in Eq. (9). It assumes the value 0 when the values of the credibility estimated by the metrics are all equal; on the other hand, it assumes the value 1 when the values estimated by one metric are equal to 0 for all workers, and the values estimated by the other metric are equal to 1 for all workers.

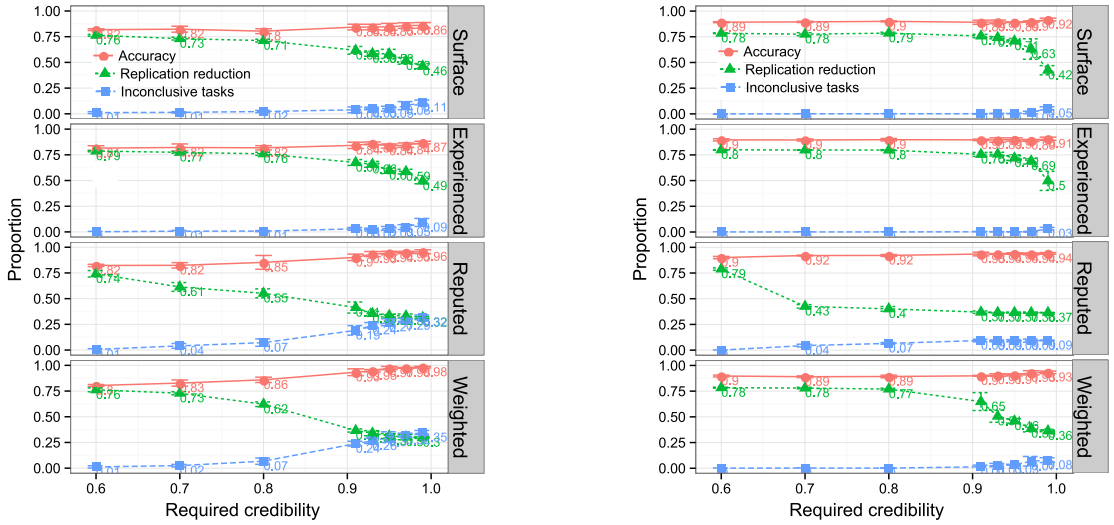
$$m(x, y) = \frac{1}{|W|} \times \sum_{w \in W} |x_w - y_w| \,. \tag{9}$$

We computed the mean absolute difference between each pair of metrics. The results indicate that *the credibility metrics tend to estimate different values of credibility*, except in very easy tasks in which the metrics estimate values of credibility closer to each other because in this case workers tend to be credible regardless of how credibility is measured. In the Fact Evaluation project, surface agreement and weighted agreement metrics are equal (distance is zero) when $d = 0.0$, $d = 0.7$, and $d = 1.0$. In the Sentiment Analysis project, surface agreement and experienced agreement metrics are equal when $d \leq 0.2$. For the other pairs of metrics, the distance is zero only when $d = 0$.

Task replication. We examine the effect of the required credibility parameter in the performance of the algorithm (Fig. 1). The main result is that *the higher the required credibility, the lower the replication reduction, and the higher both the proportion of inconclusive tasks and the accuracy on conclusive tasks*. The only benefit of increasing the required credibility is the resulting increase in accuracy. It occurs regardless the project and credibility metric used. Small differences can be observed between the metrics. For example, when the required credibility tends to 1, the metrics reputed agreement and weighted agreement tend to reach higher accuracy, but higher proportion of inconclusive tasks, and lower replication reduction compared to the other metrics. It happens because these metrics usually generate lower values of credibility than the values generated by the other metrics. Combined with a higher required credibility, it makes the algorithm less inclined to stop replication. Results in Fig. 1 are for urgency equal to 0.

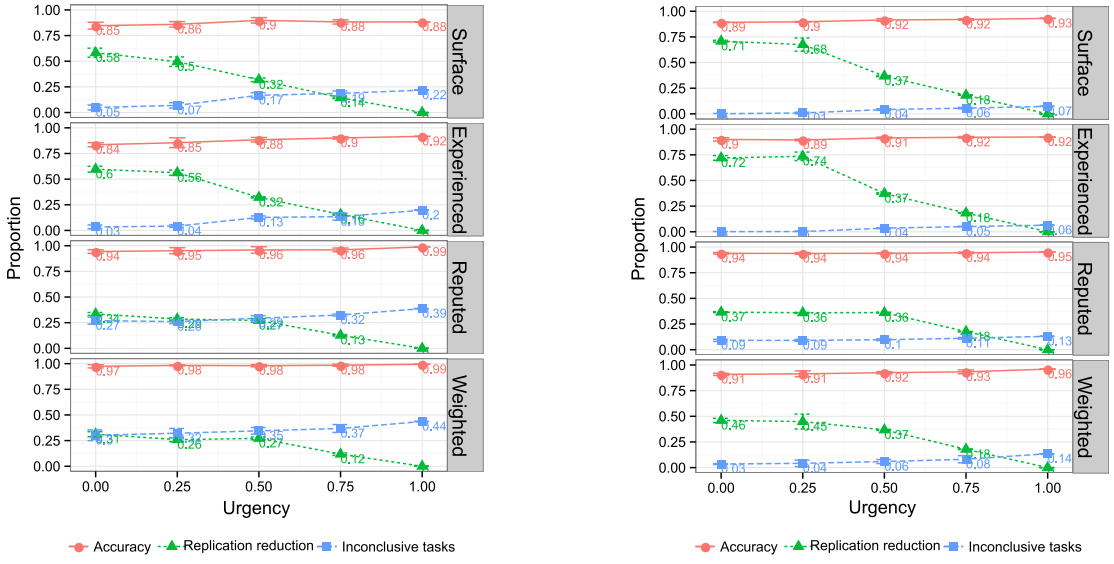
We examine the effect of the urgency parameter on the effectiveness of the task replication algorithm (Fig. 2). The most important result is that *the higher the urgency, the lower the replication reduction, and the higher both the proportion of inconclusive tasks and the accuracy*. As expected, when the urgency is 1, there is no replication reduction because all replicas of the tasks are generated and executed at once. However, one indirect benefit of using urgency 1 is that, although the algorithm has no room to reduce replication, it still chooses the final answers to the tasks according to the credibility, which improves accuracy. On the other hand, when the urgency is set to a value equal to 0, replicas of a task are generated and executed one at a time. In this case, the algorithm can detect when a credible answer is obtained and stops replication. It is the configuration in which the algorithm reaches the highest replication reduction.

We also investigate the influence that the difficulty of the tasks being replicated may have on the performance of the algorithm. Table 2 shows: (i) the correlation between the difficulty degree of a task and the credibility of the answer obtained by the algorithm ($\rho(d, c)$); and also (ii) the correlation between the difficulty degree of a task and the replication reduction achieved by the algorithm ($\rho(d, s)$). The results show a negative correlation between the



(a) Sentiment analysis. (b) Fact evaluation.

Fig. 1. Accuracy, replication reduction and proportion of inconclusive tasks generated by the proposed algorithm when the value of required credibility and the credibility metric are varied. Urgency is 0.



(a) Sentiment analysis. (b) Fact evaluation.

Fig. 2. Accuracy, replication reduction and proportion of inconclusive tasks generated by the proposed algorithm when the value of urgency and the credibility metric are varied. Required credibility is 0.95.

Table 2
Spearman correlation between the difficulty degree and credibility of answers ($\rho(d, c)$) and between the difficulty degree and replication reduction ($\rho(d, s)$).

	Sentiment analysis		Fact evaluation	
	$\rho(d, c)$	$\rho(d, s)$	$\rho(d, c)$	$\rho(d, s)$
Surface	-0.20 ± 0.06	-0.33 ± 0.05	-0.09 ± 0.01	-0.02 ± 0.01
Experienced	-0.11 ± 0.06	-0.29 ± 0.05	-0.05 ± 0.01	-0.04 ± 0.01
Reputed	-0.47 ± 0.04	-0.49 ± 0.04	-0.80 ± 0.003	-0.74 ± 0.01
Weighted	-0.44 ± 0.05	-0.50 ± 0.04	-0.16 ± 0.01	-0.04 ± 0.01

difficulty degree and the credibility of the answers. It indicates that *the higher the difficulty degree of a task, the lower the credibility of the answer obtained by the algorithm*. This correlation is more relevant when the algorithm uses the metrics reputed agreement ($\rho(d, c) = -0.47$ in the Sentiment Analysis project and $\rho(d, c) = -0.8$ in the Fact Evaluation project). The results also show that the replication reduction is negatively correlated to the difficulty degree of a task. This indicates that *the higher the difficulty degree, the lower the replication reduction achieved by the algorithm*. This correlation is strong when the metric reputed agreement is used in the Fact Evaluation project ($\rho(d, s) = -0.74$) and when the metric weighted agreement is used in the Sentiment Analysis project

($\rho(d, s) = -0.5$). Overall, these results indicate that the impact of the algorithm is more moderate in terms of replication reduction and answers credibility improvement when the tasks are difficult.

Pareto front. We analyse the best configurations of the algorithm by using the Pareto front analysis. We evaluate 160 configurations that differ among themselves in terms of required credibility, urgency and the credibility metric used. We consider both the conservative and non-conservative perspectives.

In the non-conservative perspective, the Pareto front consists of 13 (8%) configurations in the Fact Evaluation project and 26 (16%) configurations in the Sentiment Analysis project. No configuration in the Pareto front includes the metric surface agreement, indicating that such metric is dominated by the others. In the conservative perspective, in turn, the Pareto front consists of 6 (4%) configurations in the Sentiment Analysis project (Table 3) and 3 (2%) configurations in the Fact Evaluation project (Table 4). The choice of one of these configurations depends on the requester's interest, giving higher priority to the accuracy, replication reduction, or both. Only the metrics experienced agreement and reputed agreement appear in the Pareto front, being, therefore, the best metrics to optimise accuracy and replication reduction in this case. Analysing the performance of the comparative strategies (Tables 3 and 4), we observe that the configurations in the Pareto front reach accuracy, if not higher, at least comparable with majority voting, but in most cases with a substantial gain in terms of replication reduction. By construction, the accuracy reached by the oracle is the highest possible, and the replication reduction is the highest achievable, without compromising accuracy. When compared to the oracle strategy, the proposed approach achieves equivalent replication reduction, with a small decrease on the accuracy attained. Of course, in the proposed algorithm, if the requester sets a very low value for the required credibility or a very high urgency, both accuracy and replication reduction will be compromised. Thus, it is expected that some configurations of the algorithm may achieve results that are worse than those achieved by the comparative strategies.

Finally it is worth noting that even low replication reduction can be of great relevance when the absolute values are analysed. For example, we analyse the absolute values of the lowest and largest replication reduction in Tables 3 and 4. In the Sentiment Analysis project, the lowest replication reduction is 0.17, which represents a reduction of 96,794 replicas in the project and an average of 1 replica reduced per task. The largest replication reduction is 0.78, which represents a reduction of 444,112 replicas in the whole project and an average of 4 replicas reduced per task. In the Fact Evaluation project, the lowest replication reduction is 0.78, which represents a reduction of 171,600 replicas in the project and an average of 4.02 replicas reduced per task. The larger replication reduction in this project is 0.80, which represents a reduction of 176,000 replicas in the whole project and an average of 4.13 replicas reduced per task.

5. Discussion

Our results show the characteristics and performance of our approach to measure workers' credibility and perform the replication of human computation tasks considering information about the credibility score of workers. In this section we discuss the main novelties brought by our study, the implications for human computation and related fields, and assumptions and limitations of our analysis.

The performance of task replication in human computation. The effectiveness of the replication is highly influenced by both the parameters defined by the requesters (e.g. required credibility and urgency) and tasks characteristics (e.g. difficulty). When the replication is configured to prioritise the accuracy of the answers, it is able to outperform the majority voting reference strategy. When it is configured to prioritise replication reduction, it is able to outperform the oracle reference strategy. Regarding the difficulty of tasks, our algorithm is more effective in easy tasks. It autonomously finds such tasks and uses a suitable level of replication for them, obtaining credible answers and reduction of replication.

Implications. Our findings contribute to a growing body of knowledge about workers agreement. By exploring four different ways to measure agreement, and highlighting differences among them, we expand the concept of agreement, and establish different dimensions not considered previously. Together with previous work [40,47], our results suggest that the agreement among workers is the common case, while disagreement is an important signal of problems in the task. The Shannon entropy can help us to find hard tasks, but also poorly designed tasks or problem in their input data.

We show that the proposed metrics can be incorporated in the dynamics of task replication in human computation systems in order to enhance their performance. The credibility-based task replication allows requesters to obtain accurate answers and replication reduction, while they control other requirements, such as urgency and required credibility. It shows the importance of replication strategies in the design space of human computation system, which had been under explored before. Still it remains to be known whether one can get a perfect accuracy as reached by the oracle. Even statistical methods focused only on increasing accuracy have not achieved perfect accuracy as generated by the oracle [36,38]. Also, there are tasks in which only an expert can decide which answer is most appropriate [48,40].

Task replication can add to other initiatives that have been extensively investigated in human computation systems. For example, our algorithm can be combined with a task scheduling algorithm in a way that our algorithm decides to which tasks it is advantageous to get more answers from the workers and the scheduling algorithm decides the right workers to perform such tasks, e.g. the most credible ones. It works similarly to an active learning approach. Task replication can also be combined with an expert review strategy. Because experts are usually in low availability and high cost, it is more advantageous to use their computational power only for the few task that cannot be dealt with by ordinary workers. This is the case of inconclusive tasks identified by our algorithm.

An appropriate definition of the degree of replication of tasks can also benefit workers. For example, in crowdsourced citizen science systems, an excessive redundancy per task mean that the time devoted by volunteers is being wasted with unnecessary redundant work. It can be a discouraging factor for volunteers willing to contribute [49,50]. By using an appropriate degree of replication, the system ensures that the time devoted by the volunteer is being well used.

The proposed credibility metrics can be used to inform the system about how to provide contextual feedback to workers. Providing feedback to workers have shown to be very important to keep them engaged and improve their performance [51]. Such feedback can inform each worker in which aspect he/she falls short, promoting the evolution of their skills [52]. For example, the system can inform a worker that he/she has exhibited low experienced agreement, which indicates that answers provided by him/her have a strong random bias. Besides such feedback, training tasks can be routed in a way to help workers improving their credibility.

Table 3
Configurations in the Pareto front of the sentiment analysis project.

Algorithm configuration			Performance	
Required credibility	Urgency	Metric	Replication reduction	Accuracy
0.60	0.00	Experienced	0.78 ± 0.01	0.83 ± 0.02
0.70	0.00	Experienced	0.77 ± 0.01	0.84 ± 0.04
0.91	0.00	Experienced	0.66 ± 0.03	0.85 ± 0.03
0.93	0.00	Experienced	0.64 ± 0.03	0.86 ± 0.01
0.91	0.25	Reputed	0.33 ± 0.03	0.87 ± 0.04
0.70	0.75	Experienced	0.17 ± 0.01	0.89 ± 0.02
Comparative scenarios				
Majority voting with 3 replicas			0.48	0.83 ± 0.02
Majority voting with all replicas			0.00	0.86
Oracle			0.76 ± 0.01	1.00

Table 4
Configurations in the Pareto front of the fact evaluation project.

Algorithm configuration			Performance	
Required credibility	Urgency	Metric	Replication reduction	Accuracy
0.60	0.00	Experienced	0.80 ± 0.00	0.90 ± 0.01
0.70	0.00	Experienced	0.80 ± 0.00	0.90 ± 0.01
0.60	0.00	Reputed	0.78 ± 0.01	0.91 ± 0.01
Comparative scenarios				
Majority voting with 3 replicas			0.42	0.89 ± 0.01
Majority voting with all replicas			0.00	0.90
Oracle			0,74 ± 0.01	1.00

Limitations. The proposed approach has limitations that should be highlighted. It is designed for human computation projects made up of micro-tasks, not being appropriate for macro-tasks projects. The credibility metrics make no assumptions on the type of instructions of tasks, and type of input data, e.g. images, text, audio, video, and so forth. However, because they aggregate the answers to detect convergences and divergences, they assume that the answers provided by the workers are structured. This is usually the case in classification, and multiple-choice tasks. We only evaluated our approach with factual tasks because of the limited availability of data sets with non-factual tasks. However, there is nothing in the metrics and algorithms that prevent them from being used with non-factual tasks whose answers are aggregated based on agreement. Finally, the estimation of the credibility of workers per each difficulty degree may be less accurate for workers who perform few tasks. Fortunately, the largest contributions to the systems are normally given by the minority of workers who works on a large set of tasks [49,53].

6. Conclusion

In this paper, we explored the use of agreement-based credibility metrics to improve task replication in human computation systems. Our contribution is threefold: (1) we integrate concepts from the literature of human computation, namely credibility assessment and replication of tasks; (2) we propose four metrics to automatically measure the credibility of workers while they execute tasks in the system; (3) we propose an adaptive task replication algorithm that optimises the degree of replication according to workers' credibility and requesters' requirements. Our evaluation consisted in simulating the replication algorithm using data from two human computation systems, and covering a wide parameter space. The results show that the credibility-based task replication algorithm can be effective in reducing the degree of replication, while meeting other requirements from the requesters, such as required credibility and urgency.

Our study suggests a number of avenues for future work. For example, future research can investigate: (1) the use of credibility metrics to provide contextual feedback to workers about their performance; (2) the combination of credibility assessment with other mechanisms such as gold standard tasks and expert review; and (3) in-depth analysis of the role that task replication plays in the design space of human computation systems. The promising results presented in this paper should also catch the attention of the operators of existing human computation platforms. It would be of great interest to have the proposed techniques implemented and deployed in real systems, and assess their performance in these systems. Thus, research on credibility assessment and task replication promises to continue to be an exciting aspect for distributed human computation.

Acknowledgements

We are grateful to Herman Martins, Jussara Almeida and Nazareno Andrade for their suggestions to improve several aspects of the manuscript. Francisco Brasileiro is a CNPq/Brazil researcher.

References

[1] L. von Ahn, Human computation, in: 46th ACM/IEEE Design Automation Conference, IEEE, Washington, DC, USA, 2009, pp. 418–419.
[2] A.J. Quinn, B.B. Bederson, Human computation: A survey and taxonomy of a growing field, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, New York, NY, USA, 2011, pp. 1403–1412.
[3] L. Ponciano, F. Brasileiro, N. Andrade, L. Sampaio, Considering human aspects on strategies for designing and managing distributed human computation, J. Internet Serv. Appl. 5 (1) (2014) 1–15.
[4] L. von Ahn, B. Maurer, C. McMillen, D. Abraham, M. Blum, recaptcha: Human-based character recognition via web security measures, Science 321 (5895) (2008) 1465–1468.
[5] M.S. Bernstein, G. Little, R.C. Miller, B. Hartmann, M.S. Ackerman, D.R. Karger, D. Crowell, K. Panovich, Soylent: A word processor with a crowd inside, in: 23nd Annual ACM Symposium on User Interface Software and Technology, ACM, New York, USA, 2010, pp. 313–322.
[6] R. Crouser, R. Chang, An affordance-based framework for human computation and human-computer collaboration, IEEE Trans. Vis. Comput. Graphics 18 (12) (2012) 2859–2868.
[7] G. Paolacci, J. Chandler, P.G. Ipeirotis, Running experiments on amazon mechanical turk, Judgment Decis. Mak. 5 (5) (2010) 411–419.
[8] W. Mason, S. Suri, Conducting behavioral research on amazon's mechanical turk, Behav. Res. Methods 44 (1) (2012) 1–23.
[9] S.V. Rouse, A reliability analysis of mechanical turk data, Comput. Hum. Behav. 43 (0) (2015) 304–307.

[10] J.P. Cohn, Citizen science: Can volunteers do real research? BioScience 58 (3) (2008) 192–197.
[11] C.J. Lintott, K. Schawinski, A. Slosar, K. Land, S. Bamford, D. Thomas, M.J. Raddick, R.C. Nichol, A. Szalay, D. Andreescu, P. Murray, J. Vandenberg, Galaxy Zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey, Mon. Not. R. Astron. Soc. 389 (3) (2008) 1179–1189.
[12] J.L. Dickinson, J. Shirk, D. Bonter, R. Bonney, R.L. Crain, J. Martin, T. Phillips, K. Purcell, The current state of citizen science as a tool for ecological research and public engagement, Front. Ecol. Environ. 10 (6) (2012) 291–297.
[13] P.G. Ipeirotis, F. Provost, J. Wang, Quality management on amazon mechanical turk, in: ACM SIGKDD Workshop on Human Computation, ACM, New York, NY, USA, 2010, pp. 64–67.
[14] A. Carvalho, S. Dimitrov, K. Larson, How many crowdsourced workers should a requester hire? Ann. Math. Artif. Intell. (2016) 1–28 (First online).
[15] R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, Wiley, 1991.
[16] E. Law, L. von Ahn, Human computation: An Integrated Approach to Learning from the Crowd, in: Synthesis Lectures on Artificial Intelligence and Machine Learning Series, Morgan & Claypool, San Rafael, CA, United States, 2011.
[17] P. Michelucci, J.L. Dickinson, The power of crowds, Science 351 (6268) (2016) 32–33.
[18] M. Eitzel, J. Cappadonna, C. Santos-Lang, R. Duerr, S.E. West, A. Virapongse, C. Kyba, A. Bowser, C. Cooper, A. Sforzi, A. Metcalfe, E. Harris, M. Thiel, M. Haklay, L. Ponciano, J. Roche, L. Ceccaroni, F. Shilling, D. Dorler, F. Heigl, T. Kiessling, B. Davis, Q. Jiang, Citizen science terminology matters: Exploring key terms, Citizen Sci. Theory Pract. 2 (1) (2017). http://dx.doi.org/10.5334/cstp.96.
[19] P. Liu, Z. Li, Task complexity: A review and conceptualization framework, Int. J. Ind. Ergon. 42 (6) (2012) 553–568.
[20] J. Reason, Human Error, Cambridge University Press Cambridge, Cambridge, UK, 1990.
[21] B.J. Fogg, H. Tseng, The elements of computer credibility, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, New York, NY, USA, 1999, pp. 80–87.
[22] C.N. Wathen, J. Burel, Believe it or not: Factors influencing credibility on the web, J. Am. Soc. Inf. Sci. Technol. 53 (2) (2002) 134–144.
[23] S.Y. Rieh, D.R. Danielson, Credibility: A multidisciplinary framework, Annu. Rev. Info. Sci Technol. 41 (1) (2007) 307–364.
[24] J. Cohen, A coefficient of agreement for nominal scales, Educ. Psychol. Meas. 20 (1) (1960) 37–46.
[25] J.L. Fleiss, B. Levin, M.C. Paik, The measurement of interrater agreement, in: Statistical Methods for Rates and Proportions, John Wiley & Sons, Inc., New York, NY, USA, 2004, pp. 598–626.
[26] A.F. Hayes, K. Krippendorff, Answering the call for a standard reliability measure for coding data, Comm. Methods Meas. 1 (1) (2007) 77–89.
[27] P. Jalote, Fault Tolerance in Distributed Systems, Prentice Hall, New Jersey, USA, 1994.
[28] W. Cirne, F. Brasileiro, D. Paranhos, L.F.W. Góes, W. Voorsluys, On the efficacy, efficiency and emergent behavior of task replication in large distributed systems, Parallel Comput. 33 (3) (2007) 213–234.
[29] G. Kazai, J. Kamps, N. Milic-Frayling, An analysis of human factors and label accuracy in crowdsourcing relevance judgments, Inf. Retr. 16 (2) (2013) 138–178.
[30] A. Kittur, J.V. Nickerson, M. Bernstein, E. Gerber, A. Shaw, J. Zimmerman, M. Lease, J. Horton, The future of crowd work, in: 16th ACM Conference on Computer Supported Cooperative Work & Social Computing, ACM, New York, NY, USA, 2013, pp. 1301–1318.
[31] A. Kulkarni, M. Can, B. Hartmann, Collaboratively crowdsourcing workflows with turkomatic, in: ACM Conference on Computer Supported Cooperative Work, ACM, USA, 2012, pp. 1003–1012.
[32] D. Martin, B.V. Hanrahan, J. O'Neill, N. Gupta, Being a turker, in: 17th ACM Conference on Computer Supported Cooperative Work & Social Computing, ACM, NY, USA, 2014, pp. 224–235.
[33] S. Kochhar, S. Mazzocchi, P. Paritosh, The anatomy of a large-scale human computation engine, in: ACM SIGKDD Workshop on Human Computation, ACM, NY, USA, 2010, pp. 10–17.
[34] C. Eickhoff, A. de Vries, How crowdsourcable is your task? in: Workshop on Crowdsourcing for Search and Data Mining at the 4th ACM International Conference on Web Search and Data Mining, ACM, New York, NY, USA, 2011, pp. 11–14.
[35] J.M. Rzeszotarski, A. Kittur, Instrumenting the crowd: Using implicit behavioral measures to predict task performance, in: 24th Annual ACM Symposium on User Interface Software and Technology, ACM, New York, USA, 2011, pp. 13–22.
[36] A. Sheshadri, M. Lease, SQUARE: A benchmark for research on computing crowd consensus, in: 1st AAAI Conference on Human Computation and Crowdsourcing, AAAI, USA, 2013, pp. 156–164.
[37] F. Daniel, P. Kucherbaev, C. Cappiello, B. Benatallah, M. Allahbakhsh, Quality control in crowdsourcing: A survey of quality attributes, assessment techniques, and assurance actions, ACM Comput. Surv. 51 (1) (2018) 7:1–7:40.
[38] T.J. Bird, A.E. Bates, J.S. Lefcheck, N.A. Hill, R.J. Thomson, G.J. Edgar, R.D. Stuart-Smith, S. Wotherspoon, M. Krkosek, J.F. Stuart-Smith, G.T. Pecl, N. Barrett, S. Frusher, Statistical solutions for error and bias in global citizen science datasets, Biol. Cons. 173 (2014) 144–154.
[39] S. Sizov, Rescot: Reliable scheduling of social computing tasks, in: 3rd IEEE Inernational Conference on Social Computing and 3rd IEEE International Conference on Privacy, Security, Risk and Trust, IEEE, Washington, DC, USA, 2011, pp. 394–401.
[40] L. Aroyo, C. Welty, The three sides of crowdtruth, Hum. Comput. 1 (1) (2014) 31–44.
[41] C. Wagner, A. Suh, The wisdom of crowds: Impact of collective size and expertise transfer on collective performance, in: 47th Hawaii International Conference on System Sciences, IEEE, Washington, DC, USA, 2014, pp. 594–603.
[42] L. Ponciano, F. Brasileiro, G. Gadelha, Task redundancy strategy based on volunteers' credibility for volunteer thinking projects, in: 1st AAAI Conference on Human Computation and Crowdsourcing, AAAI, Palo Alto, CA, USA, 2013, pp. 60–61.
[43] L. Ponciano, F. Brasileiro, G. Gadelha, A. Furtado, Adaptive task replication strategy for human computation (in portuguese), in: 32nd Brazilian Symposium on Computer Networks and Distributed Systems, IEEE, Washington, DC, USA, 2014, pp. 249–257.
[44] C.E. Shannon, Prediction and entropy of printed english, Bell Syst. Tech. J. 30 (1) (1951) 50–64.
[45] L.F.G. Sarmenta, Sabotage-tolerance mechanisms for volunteer computing systems, Future Gener. Comput. Syst. 18 (4) (2002) 561–572.
[46] Z. Michalewicz, D.B. Fogel, How to Solve it: Modern Heuristics, second ed., Springer, Berlin, Germany, 2004.
[47] O. Alonso, C. M, M. Najork, Debugging a crowdsourced task with low inter-rater agreement, in: 15th ACM/IEEE-CS Joint Conference on Digital Libraries, ACM, NY, USA, 2015, pp. 101–110.
[48] V.S. Sheng, F. Provost, P.G. Ipeirotis, Get another label? improving data quality and data mining using multiple, noisy labelers, in: 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2008, pp. 614–622.
[49] L. Ponciano, F. Brasileiro, R. Simpson, A. Smith, Volunteers' engagement in human computation for astronomy projects, IEEE Comput. Sci. Eng. 16 (6) (2014) 52–59.
[50] L. Ponciano, F. Brasileiro, Finding volunteers' engagement profiles in human computation for citizen science projects, Hum. Comput. 1 (2) (2014) 245–264.
[51] S. Dow, A. Kulkarni, S. Klemmer, B. Hartmann, Shepherding the crowd yields better work, in: 15th ACM Conference on Computer-Supported Cooperative Work & Social Computing, ACM, NY, USA, 2012, pp. 1013–1022.
[52] B. Satzger, H. Psaier, D. Schall, S. Dustdar, Stimulating skill evolution in market-based crowdsourcing, in: S. Rinderle-Ma, F. Toumani, K. Wolf (Eds.), Business Process Management, in: Lecture Notes in Computer Science, vol. 6896, Springer Berlin Heidelberg, Germany, 2011, pp. 66–82.
[53] H. Sauermann, C. Franzoni, Crowd science user contribution patterns and their implications, Proc. Natl. Acad. Sci. 112 (3) (2015) 679–684.

Lesandro Ponciano is a Professor at the Pontifical Catholic University of Minas Gerais, Brazil. He received a B.S. degree in Information Systems from the Pontifical Catholic University of Minas Gerais, Brazil, in 2008, and an M.Sc. and a Ph.D. degrees in Computer Science from the Universidade Federal de Campina Grande, Brazil, in 2011 and 2015, respectively. His research interests are in cooperative systems and human-interaction interaction, with special emphasis on engagement and credibility assessments. He is a member of the Brazilian Computer Society. Contact him at lesandrop@pucminas.br.

Francisco Brasileiro is a Full Professor at the Universidade Federal de Campina Grande, Brazil. He received a B.S. degree in Computer Science from the Universidade Federal da Paraíba, Brazil, in 1988, an M.Sc. degree from the same University in 1989, and a Ph.D. degree in Computer Science from the University of Newcastle upon Tyne, UK, in 1995. His research interest is in distributed systems, with special emphasis on the convergence of on-demand computing and peer production systems. He is a member of the Brazilian Computer Society, the ACM, and the IEEE Computer Society. Contact him at fubica@dsc.ufcg.edu.br.