中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-09-001

Playing GWAP With Strategies

Ling-Jyh Chen, Bo-Chun Wang, Kuan-Ta Chen

February 25, 2009 || Technical Report No. TR-IIS-09-001

http://www.iis.sinica.edu.tw/page/library/LIB/TechReport/tr2009/tr09.html

Playing GWAP With Strategies

Ling-Jyh Chen, Bo-Chun Wang, Kuan-Ta Chen
Institute of Information Science, Academia Sinica
{cclljj, bcwang, ktchen}@iis.sinica.edu.tw

Abstract

“Human Computation” represents a new paradigm of applications that exploit people’s desire to be entertained by outsourcing certain steps of the computational process to the players. Such applications also produce useful metadata as a by-product. Games With A Purpose *(GWAP) demonstrate the potential of human computation to solve a variety of problems that computer computation cannot currently resolve completely. In this paper, we propose a metric, called* system gain*, for evaluating the performance of human computation systems, and also use analysis to study the properties of GWAP systems. We argue that GWAP systems should be played with strategies. Therefore, based on our analysis, we implement an* Optimal Puzzle Selection Strategy *(OPSA) to improve human computation. Using a comprehensive set of simulations, we demonstrate that the proposed OPSA approach can effectively improve the system gain of GWAP systems, as long as the number of puzzles in the system is sufficiently large.*

***Index Terms*—Games With A Purpose; Human Computation; Collaborative Tagging**

I. Introduction

In the last two decades, the Internet has undergone rapid growth in terms of its usage, population, geographic distribution, and applications. Recent surveys of worldwide Internet usage reported that, in 2008, there were more than 100 million Facebook users [4], 258 million registered YouTube users [4], and more than 16 million active subscriptions to Massively Multiplayer Online Games (MMOGs) [3]. The figures for Facebook and YouTube represent annual growth rates of 305% and 94%, respectively, over the previous year. It is evident that Internet users today want to socialize and be entertained, in addition to exploiting traditional applications, such as the WWW, FTP, and Email.

Among numerous emerging Internet applications, “Human Computation” represents a new paradigm that exploits people’s desire to be entertained by outsourcing certain steps of the computational process to the players [7, 8, 17]. In [18], Von Ahn proposed the use of human computation to create *Games With A Purpose* (GWAP), which provide entertainment and produce useful metadata as a by-product. By exploiting “human cycles” in computation, the paradigm has shown promise in solving a variety of problems, such as image annotation and common sense reasoning, which computer computation has been unable to resolve completely thus far.

Several human computation systems have been proposed in recent years [10, 12, 19, 21–24]. Among them, the ESP Game [19] was the first to successfully realize the advantages of human computation systems, and it was subsequently adopted as the Google Image Labeler [1]. It has been shown that the image labels collected through the ESP game are usually of good quality. Moreover, the game results allow more accurate image retrieval, help users block inappropriate images (e.g., pornographic content), and improve web accessibility (e.g., the labels can help visually impaired people surf web pages [5]).

In this work, we define a metric, called *system gain*, to evaluate the performance of human computation systems. The proposed metric considers two factors: the average time required for each puzzle, and the average outcomes produced by each puzzle. Both factors are critical for human computation systems, but unfortunately they do not complement each other. We believe that human computation systems should be *Played With Strategies* (PWS). Using analysis, we investigate the inner properties of the ESP, TagATune, and Verbosity games, which correspond to three types of GWAP systems defined in [20] (i.e., the output-agreement game, the input-agreement game, and the inversion-problem game respectively). Based on our analysis, we propose an *Optimal Puzzle Selection Algorithm* (OPSA) that maximizes the system gain by properly accommodating the two opposing factors. Using a set of simulations, we evaluate the proposed OPSA scheme on two widely used schemes, namely the *Random Puzzle Selection Algorithm* (RPSA) and the *Fresh-first Puzzle Selection Algorithm* (FPSA). The results demonstrate that, under the OPSA scheme, the GWAP system yields a much better system gain than under the two compared schemes. In addition, the presented analysis is simple and applicable to other human computation systems.

The remainder of this paper is organized as follows. Section II contains a review of related works on human computation systems. In Section III, we describe three GWAP systems, the ESP, TagATune, and Verbosity games, which correspond to the output-agreement game, the input-agreement game, and the inversion-problem game respectively. In Section IV, we present our analysis of the three GWAP systems; and in Section V, we compare three puzzle selection algorithms for the ESP game, namely the RPSA, FPSA, and OPSA schemes. Section VI presents a comprehensive set of simulation results, which we analyze and explain in detail. In Section VII, we consider several issues arising from this work. We then summarize our conclusions in Section VIII.

II. Background

The concept of “Human Computation” was pioneered by Luis Von Ahn and his colleagues, who created games with a purpose [18], which people play voluntarily. The games also produce useful metadata as a by-product. By taking advantage of people’s desire to be entertained, human computation has shown promise in solving some problems that computer computation cannot currently resolve completely. In recent years, a substantial and increasing amount of research effort has been invested in the area, and several human computation systems have been developed for a variety of purposes [10, 12, 19, 21–24].

The online ESP Game [19] was the first human computation system, and it was subsequently adopted as the Google Image Labeler [1]. In the system, two randomly selected players are paired to create a game, and a randomly selected image is displayed to both players simultaneously. To execute the task, the players must enter possible words to label the image until an "agreement" is reached (i.e., the same word is entered by both players). The agreed word is typically a good label for the image, and the system then displays another image as a new task to be solved. As mentioned earlier, it has been shown that the collected labels facilitate more accurate image retrieval, help users block inappropriate images (e.g., pornographic content), and improve web accessibility (e.g., the labels can help visually impaired people surf web pages [5]).

In addition to image annotation, the Peekaboom system [24] can help determine the location of objects in images; and the Squigl system [2] and the LabelMe system [12] provide complete outlines of the objects in an image. Phetch [21, 22] provides image descriptions that improve web accessibility and image searches, while the Matchin system [2] helps image search engines rank images based on which ones are the most appealing. The concept of the ESP Game has been applied to other problems. For instance, the TagATune system [9], which provides annotation for sounds and music, can improve audio searches. The Verbosity system [23] and the Common Consensus system [10] collect "common-sense" knowledge that is valuable for commonsense reasoning and enhancing the design of interactive user interfaces. The Context-Aware Recognition Survey (CARS) system [25] uses ubiquitous sensors to monitor activities in the home, while the Gopher system [6] employs mobile social gaming for geospatial tagging. Moreover, [15] applies human computation to ontology alignment and web content annotation for the Semantic Web using various games, such as OntoPronto, SpotTheLinks, OntoTube, and OntoBay. Finally, Shenoy and Tan [14] show that it is possible to design environments in which humans cannot avoid processing some tasks (and still produce some useful outcomes), even though they are not actively trying to do so.

III. Game Descriptions

In this section, we describe the ESP game [19], the TagATune game [9], and the Verbosity game [23], which correspond, respectively, to the output-agreement game, the input-agreement game, and the inversion-problem game defined by Von Ahn and Dabbish [20].

A. The Output-agreement Game: *ESP*

The ESP Game [19] was the first human computation system to exploit people's desire to be entertained, and provide useful metadata as a by-product. When a user logs into the system, he/she is automatically matched with a random partner. The two players do not know each other's identity as they cannot communicate.

Initially, a randomly selected image is displayed to both players simultaneously. The players then input possible words to label the image until an "agreement" is reached (i.e., the same word is entered by both players), and a bonus score is awarded to each player based on the '*quality*' of the agreed word. In practice, the '*quality*' of a word is measured by its popularity; generally, words that are more popular receive lower scores. After the players agree on a word, they are shown another image. In each game, they have two and a half minutes to label 15 images.

The word on which the two players agree becomes the label of the image, and it can not be used the next time that image is displayed in another game (the word is called a "taboo" word of the image). The rationale for using taboo words is to ensure that each image is labeled with a variety of words.

B. The Input-agreement Game: *TagATune*

The TagATune Game [9] is an input-agreement game that provides useful metadata of sounds and music (collectively referred to as *tunes*) by applying the concept of *games with a purpose*. Similar to the ESP game, when a user logs into the system, he/she is automatically matched with a partner, who is selected randomly and anonymously from a pool of available players. Again, the two players do not know each other's identity as they cannot communicate. They are asked to collaborate and determine whether they have been given the same input puzzle (i.e., tune).

In a game round, each player is given an input (tune), but only the system knows whether the inputs are the same or different. The players have to use possible words to properly describe their input. They win a game round (and obtain points) if they both correctly determine whether they have the same input tune; otherwise, they lose the game round (i.e., their score is zeros). During a game round, the players are allowed to stop or replay their respective tunes at any time, and they are allowed to see each other's outputs. Moreover, they can decide to pass over a tune if they both think it is too difficult.

Since the players want to win as many points as possible, it is in their best interest to provide accurate outputs that appropriately describe their respective inputs, so that they can determine if their inputs are the same. As a result, the words collected in a TagATune game are typically good enough to describe the input tunes. Moreover, the outcomes may become official tags of the tunes if enough people agree (the threshold of which depends on the game's statistics).

C. The Inversion-problem Game: *Verbosity*

The Verbosity Game [23] is a popular inversion-problem game that employs human computation to collect commonsense reasoning related to a word by asking one of the players to guess the input puzzle (i.e., a word) based on the other player's description of the puzzle. Like the ESP and TagATune games, two players are selected at random to create a game; once again, they do not know each other. The players take turns to play the roles of "*Describer*" and "*Guesser*". The *Describer* is asked to provide a number of words to describe the given input, and the *Guesser* has to guess the input based on the *Describer*'s outputs.

To make the game easier, Verbosity provides the *Describer* with a set of sentence templates to describe the input puzzle (e.g., "*it looks like* ______" and "*it is a type of* ______"). The *Describer* can see all of the *Guesser*'s inputs, so he can adjust his playing strategy during the game round. Since both players have to collaborate to win the game, the *Describer* must do his best to help the *Guesser* guess the input. Since the game structure encourages players to enter correct information, the collected descriptions are typically good enough to be used as official tags of the corresponding puzzles.

The inversion-problem game can be regarded as a special case of the input-agreement game, except that (1) the number of outcomes per game round is limited (i.e., there are at most m outcomes in each game round, and m is much smaller than the total number of possible outcomes); and (2) a game round cannot be a failure in inversion-problem games (i.e., the two players will pass over a puzzle if they can not complete it using the first m descriptions). For instance, in the Verbosity game, there are only 6 sentence templates (i.e., $m = 6$).

IV. Game Analysis

For the sake of efficiency, the GWAP system tries to collect outcomes with the largest possible aggregated score in the shortest possible time, for each puzzle. There is a trade-off between these two factors. On the one hand, to minimize the time required per puzzle, the system prefers that each puzzle is played only once. The rationale is to maximize the number of games that players try to solve. On the other hand, the system prefers to take as many labels as possible for each puzzle, which results in the playing of fewer distinct puzzles. Thus, an optimal puzzle selection strategy that can accommodate the two goals is highly desirable. To this end, we propose a metric to evaluate the system gain of the GWAP system, and analyze the puzzle selection problem. We discuss the analysis in the following subsections.

A. The Output-agreement Game: *ESP*

Let N be the number of puzzles that have been played at least once in the system, T be the average time consumed per puzzle, and S be the total score of all the agreed labels. We define the system gain, G, of the ESP game as follows:

$$\begin{aligned} G &= ln(C_1 \times \frac{N}{NT}) \times ln(C_2 \times \frac{S}{N}) \\ &= ln(C_1 \times \frac{1}{T}) \times ln(C_2 \times \frac{S}{N}), \end{aligned} \tag{1}$$

where C_1 and C_2 are two scaling constants that ensure $ln(C_1 \times \frac{1}{T}) > 0$ and $ln(C_2 \times \frac{S}{N}) > 0$ respectively.

Clearly, the system gain increases as the average time required per puzzle (i.e., T) decreases, and/or as the average total score (per puzzle) (i.e., $\frac{S}{N}$) increases. Suppose that, in the system, each puzzle has the potential to yield K labels in total; that each tag is associated with one positive score value based on its popularity. For simplicity, we assume there are totally X distinct scores (i.e., $S_1, S_2, S_3, ..., S_X$) in the system; and that $S_i = e^i$. Moreover, we assume that K_i labels have the score S_i, and that $K_i = e^{X-i}$. Therefore, the total number of potential labels per puzzle (K) can be derived by Eq. 2, and the expected score of each tag ($E[S]$) can be obtained by Eq. 3.

$$K = \sum_{i=1}^{X} K_i = \frac{e^X - 1}{e - 1} \tag{2}$$

$$E[S] = \frac{\sum_{i=1}^{X} e^i e^{X-i}}{K} = \frac{e^X(e-1)X}{e^X - 1} \tag{3}$$

For each game round, let P_s and $\overline{t_s}$ be the probability and the average time spent when the game round results in an agreement; P_p and $\overline{t_p}$ be the probability and the average time spent when the players decide to pass over the assigned puzzle; and P_t and $\overline{t_t}$ be the probability and the average time spent when the game round is terminated due to timeout. Since we know that $P_s + P_p + P_t = 1$, we can obtain $\bar{t}$, i.e., the expected time interval between any two consecutive outcomes in the ESP game, by

$$\bar{t} = \overline{t_s} + \frac{P_p}{P_s} \times \overline{t_p} + \frac{P_t}{P_s} \times \overline{t_t}. \tag{4}$$

Suppose each puzzle has accumulated r agreements on average (i.e., $T = r \times \bar{t}$, and $E[S] \times r = \frac{S}{N}$). We can then rewrite Eq. 1 as follows:

$$\begin{aligned}
G &= ln(C_1 \times \frac{1}{r \times \bar{t}}) \times ln(C_2 \times E[S] \times r) \\
&= \big(ln(C_1/\bar{t}) - ln(r))\big) \times (ln(C_2 E[S]) + ln(r)) \\
&= -(ln(r))^2 + (ln(C_1/\bar{t}) - ln(C_2 E[S]))ln(r) \\
&\quad + ln(C_1/\bar{t})ln(C_2 E[S]) \\
&= -\left(ln(r) - \frac{ln(C_1) - ln(\bar{t}) - ln(C_2 E[S])}{2}\right)^2 + C,
\end{aligned} \tag{5}$$

where C is a constant with a value equal to $ln(C_1/\bar{t})ln(C_2 E[S]) + \left(\frac{ln(C_1) - ln(\bar{t}) - ln(C_2 E[S])}{2}\right)^2$. Note that C also represents the largest possible system gain, which occurs when

$$r = e^{\frac{ln(C_1) - ln(\bar{t}) - ln(C_2 E[S])}{2}}. \tag{6}$$

B. The Input-agreement Game: *TagATune*

Similar to the analysis in subsection IV-A, let N be the number of puzzles that have been played at least once in the system, T be the average time required to solve each puzzle, and O be the total score of all the game outcomes. We define the system gain, G, of the TagATune game as follows:

$$\begin{aligned}
G &= ln(C_1 \times \frac{N}{NT}) \times ln(C_2 \times \frac{O}{N}) \\
&= ln(C_1 \times \frac{1}{T}) \times ln(C_2 \times E[O]),
\end{aligned} \tag{7}$$

where C_1 and C_2 are two scaling constants that ensure $ln(C_1 \times \frac{1}{T}) > 0$ and $ln(C_2 \times \frac{S}{N}) > 0$ respectively, and $E[O]$ is the expected total score of the game outcomes for each puzzle.

Clearly, the system gain increases as the average time required per puzzle (i.e., T) decreases, and/or as the average total score per puzzle (i.e., $E[O]$) increases. Suppose that each puzzle in the system has the potential to yield a total of K outcomes, each of which is associated with one positive score based on the outcome's popularity. In addition, suppose the score of the k-th outcome is v_k, and the probability that the k-th outcome will be output is p_k. For simplicity, we assume there are totally X distinct scores (i.e., $S_1, S_2, S_3, ..., S_X$) in the system, and that $S_i = e^i$. We also assume that K_i outcomes have the score S_i, and that $K_i = e^{X-i}$ (i.e., the total number of potential outcomes per puzzle (K) derived by Eq. 2. Then, we can obtain the expected total score of the new outcomes (per puzzle) in the r-th game round, $E[O_r]$, by Eq. 8, and the expected total score of all the outcomes after the first r game rounds, $E[O]$, by Eq. 9.

$$\begin{aligned}
E[O_r] &= (1 - p_1)^{r-1} \times p_1 \times v_1 + (1 - p_2)^{r-1} \times p_2 \times v_2 \\
&\quad + \ldots + (1 - p_K)^{r-1} \times p_K \times v_K \\
&= \sum_{k=1}^{K} (1 - p_k)^{r-1} \times p_k \times v_k.
\end{aligned} \tag{8}$$

$$\begin{aligned} E[O] &= \sum_{i=1}^{r} E[O_i] \\ &= \sum_{i=1}^{r}\sum_{k=1}^{K} (1-p_k)^{i-1} \times p_k \times v_k \\ &= \sum_{k=1}^{K} p_k \times v_k \times \sum_{i=1}^{r} (1-p_k)^{i-1} \\ &= \sum_{k=1}^{K} p_k \times v_k \times \frac{1-(1-p_k)^r}{1-(1-p_k)} \\ &= \sum_{k=1}^{K} v_k(1-(1-p_k)^r). \end{aligned} \tag{9}$$

For each game round, let P_s and $\overline{t_s}$ be, respectively, the probability and the average time spent when the game round results in an agreement; P_p and $\overline{t_p}$ be, respectively, the probability and the average time spent when the players decide to pass over the assigned puzzle; P_t and $\overline{t_t}$ be, respectively, the probability and the average time spent when the game round is terminated due to timeout; and P_f and $\overline{t_f}$ be the probability and the average time spent when the game round fails. Since we know that $P_s + P_p + P_t + P_f = 1$, we can obtain $\bar{t}$, i.e., the expected time interval between any two consecutive outcomes in the TagATune game, by

$$\bar{t} = \overline{t_s} + \frac{P_p}{P_s} \times \overline{t_p} + \frac{P_t}{P_s} \times \overline{t_t} + \frac{P_f}{P_s} \times \overline{t_f}. \tag{10}$$

In total, there are X distinct scores (i.e., e, e^2, e^3, ..., e^X), e^{X-i} outcomes with scores equal to e^i, and the occurrence probabilities are equal to $1/e^i$; hence, we know that

$$\begin{aligned} \sum_{k=1}^{K} v_k(1-(1-p_k)^r) &\approx \sum_{k=1}^{K} v_k(1-(1-rp_k)) \\ &= r\sum_{k=1}^{K} v_k p_k = rK. \end{aligned} \tag{11}$$

Moreover, since $T = r \times \bar{t}$, we can rewrite Eq. 7 as follows:

$$\begin{aligned} G &= ln(C_1 \times \frac{1}{r \times \bar{t}}) \times ln(C_2 \times \sum_{k=1}^{K} v_k(1-(1-p_k)^r)) \\ &\approx \left(ln(\frac{C_1}{\bar{t}}) - ln(r)\right) \times (ln(C_2) + ln(rK)) \\ &= \left(ln(\frac{C_1}{\bar{t}}) - ln(r)\right) \times (ln(C_2K) + ln(r)) \\ &= -\left(ln(r) - \frac{ln(\frac{C_1}{\bar{t}}) - ln(C_2K)}{2}\right)^2 + C, \end{aligned} \tag{12}$$

where C is a constant with a value equal to $ln(\frac{C_1}{\bar{t}})ln(C_2K) + \left(\frac{ln(\frac{C_1}{\bar{t}})-ln(C_2K)}{2}\right)^2$. Note that C also represents the largest possible system gain, which occurs when

$$r = e^{\frac{ln(\frac{C_1}{\bar{t}})-ln(C_2K)}{2}}. \tag{13}$$

C. The Inversion-problem Game: *Verbosity*

We consider that the inversion-problem game is a special case of input-agreement games, except for the following factors.

1) There are no game round failures in an inversion-problem game (i.e., $P_f = 0$). Thus, we can rewrite Eq. 10 as

$$\bar{t} = \overline{t_s} + \frac{P_p}{P_s} \times \overline{t_p} + \frac{P_t}{P_s} \times \overline{t_t}. \tag{14}$$

2) The number of possible outcomes per game round is limited (i.e., m is limited); however, Equations 8 and 9 still hold.

Therefore, similar to input-agreement games, we can obtain the system gain by

$$\begin{aligned} G &= ln(C_1 \times \frac{N}{NT}) \times ln(C_2 \times \frac{O}{N}) \\ &\approx -\left(ln(r) - \frac{ln(\frac{C_1}{\bar{t}}) - ln(C_2K)}{2}\right)^2 + C, \end{aligned} \tag{15}$$

where C is a constant with a value equal to $ln(\frac{C_1}{t})ln(C_2K) + \left(\frac{ln(\frac{C_1}{t}) - ln(C_2K)}{2}\right)^2$. Note that C also represents the largest possible system gain, which occurs when

$$r = e^{\frac{ln(\frac{C_1}{t}) - ln(C_2K)}{2}}. \tag{16}$$

V. Game Strategies

In this section, we present three puzzle selection algorithms for the GWAP system, namely the *Random Puzzle Selection Algorithm* (RPSA), the *Fresh-first Puzzle Selection Algorithm* (FPSA), and the proposed *Optimal Puzzle Selection Algorithm* (OPSA). We take the RPSA scheme's performance as the baseline (in terms of system gain). The heuristics-based FPSA scheme tries to maximize the first component of Equations 1, 7, and 15 (i.e., minimize the value of T); and the OPSA scheme tries to achieve the largest possible system gain based on our analysis (as discussed in Sec. IV).

We use P to denote the set of all puzzles in the system, and we define the following three functions used by the puzzle selection algorithms: 1) $Select\_Random(P)$, which randomly selects a puzzle from the input puzzle set P; 2) $Select\_Played(P)$, which selects the puzzle from the input puzzle set P that has been played most frequently; and 3) $Select\_Fresh(P)$, which selects the puzzle from the input puzzle set P that has been played least frequently. We discuss the three algorithms in the following.

A. RPSA and FPSA

Algorithms 1 and 2 detail the steps of the *Random Puzzle Selection Algorithm* (RPSA) and the *Fresh-first Puzzle Selection Algorithm* (FPSA) respectively. In each round, RPSA selects a puzzle at random from the puzzle pool P. FPSA, on the other hand, selects the puzzle that has been played least frequently in the system. It is a greedy, heuristics-based approach that tries to maximize the first component of Eq. 1.

Algorithm 1 The Random Puzzle Selection Algorithm (RPSA).

```
Function RPSA
$p \Leftarrow Select\_Random(P)$
Return $p$
```

Algorithm 2 The Fresh-first Puzzle Selection Algorithm (FPSA).

```
Function FPSA
$p \Leftarrow Select\_Fresh(P)$
Return $p$
```

B. The Proposed Scheme: OPSA

In the proposed Optimal Puzzle Selection Algorithm (OPSA), N denotes the number of puzzles that have been played, and E denotes the expected score of each label. In addition, r denotes the optimal number of rounds (discussed in Sec. IV); and for each entry p of P, $p.r$ represents the round number in which the puzzle p was played. Suppose that the puzzle set P_0 contains all the puzzles that have not been played; P_1 contains all the puzzles that have been played at least once, but less than r rounds; and the set $P_2 = P - P_0 - P_1$ contains the other puzzles. We detail the OPSA algorithm in Algorithm 3.

Algorithm 3 The Optimal Puzzle Selection Algorithm (OPSA).

```
Function OPSA
if $\{P_1\}$ is NOT empty then
  $p \Leftarrow Select\_Played(P_1)$
  $p.r \Leftarrow p.r + 1$
  if $p.r = r$ then
    Move $p$ from $P_1$ to $P_2$
  end if
  Return $p$
else
  if $\{P_0\}$ is NOT empty then
    $p \Leftarrow Select\_Random(P_0)$
    $p.r \Leftarrow 1$
    if $p.r < r$ then
      Move $p$ from $P_0$ to $P_1$
    else
      Move $p$ from $P_0$ to $P_2$
    end if
    Return $p$
  else
    $p \Leftarrow Select\_Fresh(P_2)$
    $p.r \Leftarrow 1$
    Return $p$
  end if
end if
```

VI. Evaluation

A. Output-agreement Games

In this section, we describe the simulations performed to investigate the intrinsic properties of the ESP game based on our analysis. We also evaluate the system gain of the three puzzle selection strategies. For simplicity, we set the values of the two scaling constants C_1 and C_2 to 200 and 1 respectively. Moreover, we set the values of the parameters $(\overline{t_s}, \overline{t_p}, \overline{t_t})$ to (0.2, 0.4, 0.15)[1]; and the parameters (P_s, P_p, P_t) to (0.6, 0.3, 0.1)[2]. All the results are based on the average performance of 100 simulations.

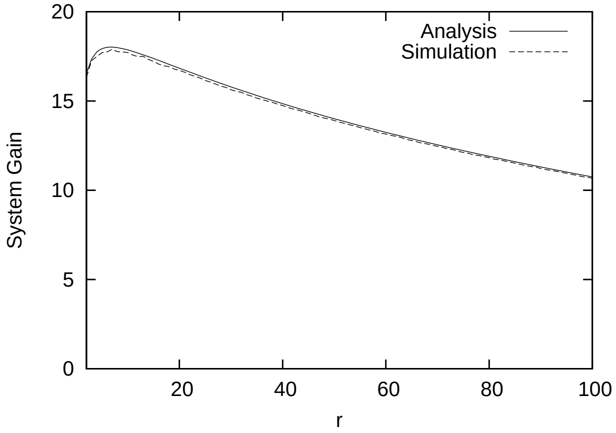

Fig. 1. Comparison of the system gain under various r settings in both the simulations and the analysis. ($X = 6$)

1) The Optimal r: In the first set of simulations, we evaluated our analytical model's accuracy in determining an optimal r value for the ESP game. We assumed that the number of puzzles in the system was infinite, and that all of them were unsolved at the beginning of the simulation (i.e., no labels were discovered for any puzzles). Figure 1 shows the evaluation results in terms of the system gain for r values between 2 and 100, when the maximum score value X was fixed at 6. In the figure, the analysis curve is derived by Eq. 5, where $E[S]$ can be obtained by Eq. 3. The optimal r is equal to 10.3353 when $X = 6$. We observe that the analysis curve matches the simulation curve very well, and the optimal r values (i.e., those that yielded the largest system gain) of the two curves are also comparable.

Next, we varied the maximum score value X in the range 2 to 10 and compared the derived optimal r values using both simulations and analysis, as shown in Figure 2. The results indicate that the analysis curve only matches

[1] Intuitively, $\overline{t_p} > \overline{t_s}$, since the players tend to pass over a puzzle when they realize it will be difficult to reach an agreement. Moreover, the players may be forced to terminate a game round simply because they do not have enough time to finish the puzzle, i.e., $\overline{t_s} > \overline{t_t}$.

[2] It is important to ensure that $P_s > P_p$; otherwise, the players may feel frustrated when playing the game. Moreover, P_t must be very small so that the players have sufficient time to solve the puzzles in order to increase the system efficiency.

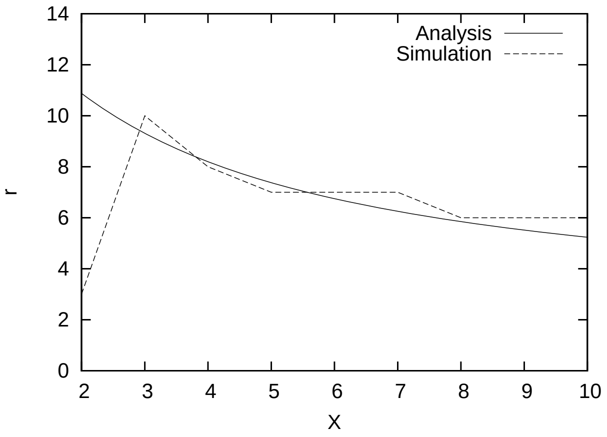

Fig. 2. Comparison of the optimal r values derived by simulations and analysis, where X varies between 2 and 10.

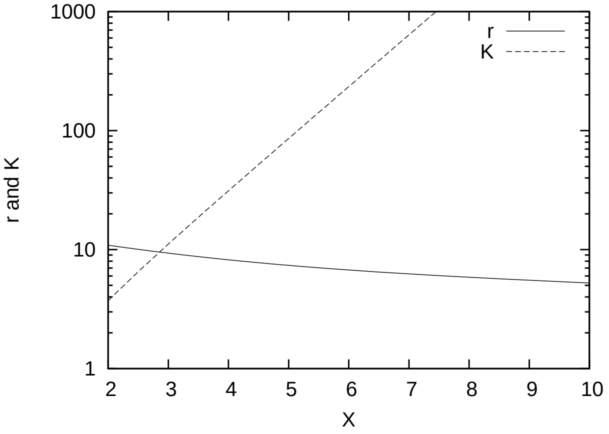

Fig. 3. The relationship between the values of r and K for various X values.

the simulation results well when $X \geq 3$. The reason is that, in the analytical model, the optimal r value is larger than the total number of potential tags per puzzle (K in Eq. 2) when $X < 3$, as shown in Figure 3. Thus, the model can not be used when $X < 3$ because the optimal number of rounds per puzzle (r) is larger than the number of tags (K) that a puzzle has in the system.

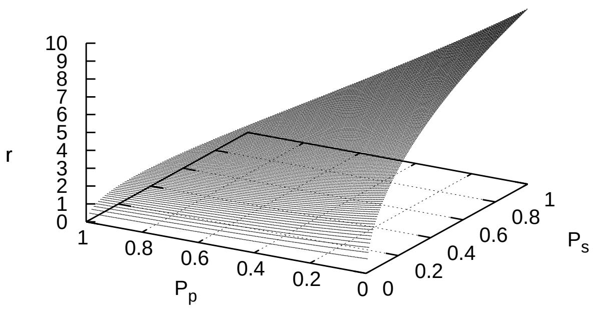

Fig. 4. The relationships between the values of r and the different values of P_s and P_p, where $X = 6$ and $P_t = 1 - P_s - P_p$.

Figure 2 shows that, when $X > 3$, the optimal r value decreases as the X value increases. This confirms our intuition that the value of $E[S]$ increases as X increases (cf. Eq. 3). As a result, based on Eq. 6, the optimal r value will decrease as X increases. We find that if there are several different scores in the system, more rounds of each puzzle must be played in order to achieve a better overall system gain.

In addition, we evaluate the relationship between the system parameters (i.e., P_s, P_p, and P_t) and the number of game rounds required to maximize the system gain r in the proposed Optimal Puzzle Selection Algorithm. Figure 4 shows the comparison results of r and different values of P_s and P_p (X is fixed at 6 and $P_t = 1 - P_s - P_p$). From the results, we find that the values of r increase with the values of P_s; whereas, given the same value of P_s, r is more resilient against the changes in P_p and P_t values. In other words, the optimal r value is highly correlated to the probability of achieving an agreement in a game round. Moreover, the higher the likelihood of reaching an agreement in each game round, the larger value of r will be.

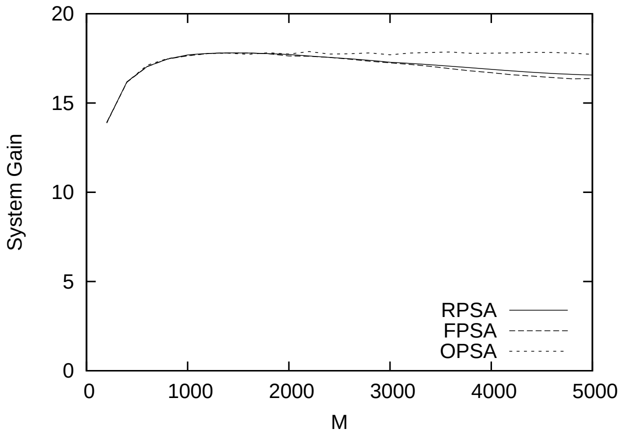

Fig. 5. Comparison of the system gain achieved by the OPSA, FPSA, and RPSA schemes with various numbers of puzzles, where X is set to 6 and the system gain is calculated after 10,000 agreements are reached in each simulation run.

2) Comparison of RPSA, FPSA, and OPSA: We now evaluate the performance of the three puzzle selection algorithms on the ESP game in terms of the system gain. In the simulation, we set $X = 6$ and calculate the system gain after 10,000 agreements are reached in each simulation run. Let M denote the total number of puzzles in the system. The simulation results are shown in Figure 5.

The results in Figure 5 show that when M is small (say, smaller than a threshold M'), the three algorithms are comparable in terms of the system gain achieved. However, when M is larger than M', the system gain of OPSA remains constant, regardless of the changes in the value of M. In contrast, the system gain of FPSA and RPSA degrades as the value of M increases, although RPSA degrades slightly less than FPSA when M is very large. More precisely, the threshold M' represents the minimal number of puzzles required to achieve the maximum system gain (i.e., $M' = N = 10000/r$). Since $X = 6$, we know that $E[S] = 10.3353$ and $r = 7$ (from Eq. 3 and 6 respectively). Therefore, in this case, $M' = 10000/7 \approx 1429$. The results indicate that, under the OPSA scheme, the ESP game must maintain at least a certain number of puzzles to achieve the maximum system gain[3]; otherwise, it will favor the RPSA and FPSA schemes because their performance is comparable to that of OPSA and they are relatively easy to implement.

B. Input-agreement Games and Inversion-problem Games

Next, we discuss the simulations performed to investigate the inner properties of the TagATune game. We also evaluate the system gain of the three puzzle selection strategies. For simplicity, we set the values of the two scaling constants C_1 and C_2 to 200 and 1 respectively. Moreover, we set the values of the parameters $(\overline{t_s}, \overline{t_f}, \overline{t_p}, \overline{t_t})$ to (0.2, 0.2, 0.4, 0.15)[4], and the parameters (P_s, P_f, P_p, P_t) to (0.5, 0.3, 0.1, 0.1)[5]. All the results are based on the average performance of 100 simulations.

1) The Optimal r: In the first set of simulations, we evaluated the accuracy of our analytical model in determining an optimal r value for the TagATune game. We assumed that the number of puzzles in the system was infinite, and all of them were unsolved at the beginning of the simulation (i.e., no labels were discovered for any

[3] Fortunately, this is not usually a problem, since the number of the puzzles can be easily increased by adding new puzzles from the Internet.

[4] Intuitively, $\overline{t_p} > \overline{t_s}$, since the players tend to pass a puzzle when they realize it will be difficult to reach an agreement; and $\overline{t_s} = \overline{t_f}$, since they are both equal to the average time required for the players to reach an agreement. Moreover, the players may be forced to terminate a game round simply because they do not have enough time to finish the puzzle, i.e., $\overline{t_s} > \overline{t_t}$.

[5] It is important to ensure that $P_s > P_p$, otherwise the players may feel frustrated when playing the game. Moreover, P_t must be very small so that the players have sufficient time to solve the puzzles in order to increase the system efficiency.

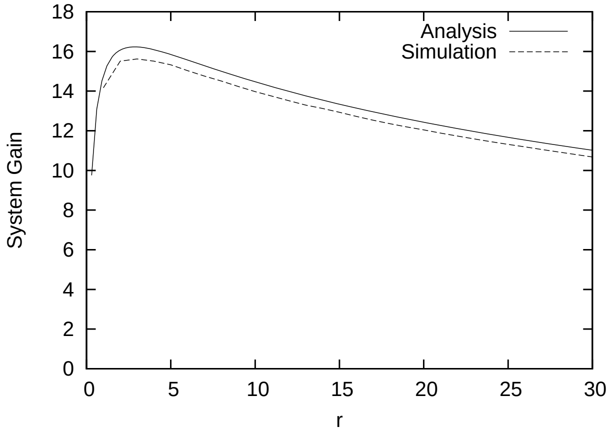

Fig. 6. Comparison of the system gain under various r settings in both the simulations and the analysis. ($X = 3$)

puzzles). Figure 6 shows the evaluation results in terms of the system gain for r values between 2 and 30 when the maximum score value X was fixed at 3. In the figure, the analysis curve, which is derived by Eq. 12, matches the simulation curve, and the optimal r values (i.e., those that yielded the largest system gain) of the two curves are also comparable.

We also evaluated the relationships between the system parameters (i.e., P_s, P_p, P_t, and P_f) and the number of game rounds required to maximize the system gain r under OPSA. Figure 7 shows the comparison results for r and different values of P_s and P_p (X is fixed at 3, $P_p = 0$, and $P_t = 1 - P_s - P_f$). From the results, we observe that the value of r increases with the values of P_s; whereas, given the same value of P_s, r is more resilient against the changes in P_p and P_t values. Clearly, the optimal r value is highly correlated to the probability of achieving an agreement in a game round; that is, the higher the probability, the larger value of r will be.

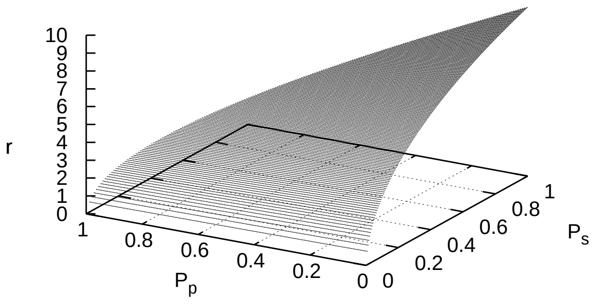

Fig. 7. The relationships between the values of r and different values of P_s and P_p, where $X = 3$, $P_p = 0$, and $P_t = 1 - P_s - P_f$.

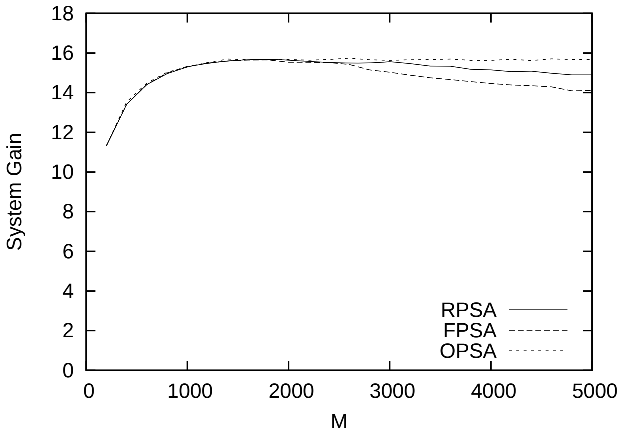

Fig. 8. Comparison of the system gain achieved by the OPSA, FPSA, and RPSA schemes with various numbers of puzzles, where X is set to 3 and the system gain is calculated after 5,000 agreements are reached in each simulation run.

2) Comparison of RPSA, FPSA, and OPSA: Here, we evaluate the three puzzle selection algorithms on input-agreement and inversion-problem games. In the simulation, we set $X = 3$ and calculate the system gain after 5,000 agreements are reached in each simulation run. Let M denote the total number of puzzles in the system. The simulation results are shown in Figure 8.

The results in Figure 8 show that when M is small (say, smaller than a threshold M'), the three algorithms are comparable in terms of the system gain achieved. However, when M is larger than M', the system gain of OPSA remains constant regardless of the changes in the values of M. In contrast, the system gain of FPSA and RPSA degrades as the value of M increases, although RPSA degrades slightly less than FPSA when M is very large. More precisely, the threshold M' represents the minimal number of puzzles required to achieve the maximum system gain (i.e., $M' = N = 5000/r$). Since $X = 3$, we know that $r = 3$ (by Eq. 13). Therefore, in this case, $M' = 5000/3 \approx 1667$. The results indicate that, under the OPSA scheme, the TagATune game must maintain at least a certain number of puzzles to achieve the maximum system gain[6]; otherwise, it will favor the RPSA and FPSA schemes because their performance is comparable to that of OPSA and they are easy to implement.

VII. Discussion

We have presented an analytical model for evaluating the system gain of three types of GWAP systems. In addition, we have proposed the Optimal Puzzle Selection Algorithm (OPSA), which strategically selects the puzzles that should be played next in order to achieve the largest possible system gain. However, there are some issues that have yet to be addressed. Below, we briefly discuss those issues and suggest possible solutions.

First, the proposed analytical model is only valid for two-player GWAP systems. However, recent studies have reported that multi-player games should improve the performance (i.e., the efficiency and quality) of future GWAP systems [11]. Therefore, extending the proposed model to support multi-player GWAP systems as well as other emerging types of GWAP games (e.g., mobile GWAP systems) would be beneficial.

Second, in this study, we assume that each game round produces an outcome with a positive score. However, it is possible for a round to have a negative outcome (e.g., if incorrect labels are assigned either intentionally or accidentally). Thus, we need a validation mechanism to examine the outcomes of games [16]. Moreover, it is necessary to extend the proposed analytical model to consider scenarios of zero and negative outcomes for more general cases. We defer a detailed evaluation of this issue to a future work.

Finally, our current analytical model does not consider the factor of *player diversity*, such as the player's age, gender, interests, language proficiency, education, and occupation, in the design of GWAP systems. Since it is widely accepted that diversity is a key factor in improving efficiency, productivity, and overall success [13], GWAP systems would benefit substantially by embracing the diversity of players. To this end, it is necessary to combine research on human computation and social networks, and design a set of *player selection strategies* to better utilize player diversity in GWAP systems. Again, we leave a detailed discussion of this issue to future work.

VIII. Conclusion

In this paper, we have studied GWAP systems, which represent an emerging human computation paradigm, and

[6] Fortunately, this is not usually a problem, since the number of the puzzles can be easily increased by adding new puzzles from the Internet.

proposed a metric called system gain for evaluating a game's performance. Moreover, we argue that GWAP systems need to be *played with strategies* in order to collect human intelligence in a more efficient manner. Based on our analysis, we propose and implement the Optimal Puzzle Selection Algorithm (OPSA) to provide guidelines for improving GWAP systems. Using a comprehensive set of simulations, we investigated the properties of GWAP systems, and demonstrated that the proposed OPSA scheme substantially outperforms other schemes in all test cases. Furthermore, the proposed analysis is simple and applicable to other human computation games, and the proposed puzzle selection strategy shows promise for use in the design and implementation of future GWAP systems.

Acknowledgement

This research was supported in part by Taiwan E-learning and Digital Archives Programs (TELDAP) sponsored by the National Science Council of Taiwan under NSC Grants: NSC 98-2631-H-001-013 and NSC 97-2631-S-003-003.

References

[1] Google Image Labeler. http://images.google.com/imagelabeler/.

[2] GWAP. http://www.gwap.com/gwap/.

[3] MMOGCHART. http://www.mmogchart.com/.

[4] TechCrunch. http://www.techcrunch.com/.

[5] J. P. Bigham, R. S. Kaminsky, R. E. Ladner, O. M. Danielsson, and G. L. Hempton. WebInSight: making web images accessible. In *ACM SIGACCESS Conference on Assistive Technologies*, pages 181–188, 2006.

[6] S. Casey, B. Kirman, and D. Rowland. The gopher game: a social, mobile, locative game with user generated content and peer review. In *International Conference on Advances in Computer Entertainment Technology*, pages 9–16, 2007.

[7] J. Howe. The Rise of Crowdsourcing. *WIRED Magazine*, 14(6), June 2006.

[8] A. Koblin. The Sheep Market: Two Cents Worth. Master's thesis, UCLA, 2006.

[9] E. L. M. Law and L. von Ahn. Input-Agreement: A New Mechanism for Data Collection Using Human Computation Games. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, 2009.

[10] H. Lieberman, D. Smith, and A. Teeters. Common Consensus: a web-based game for collecting commonsense goals. In *ACM Workshop on Common Sense for Intelligent Interfaces*, 2007.

[11] C.-W. Lin, K.-T. Chen, L.-J. Chen, I. King, and J. H.-M. Lee. An Analytical Approach to Optimizing The Utility of ESP Games. In *IEEE/WIC/ACM International Conference on Web Intelligence*, 2008.

[12] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision*, 77(1-3):157–173, May 2008.

[13] S. Scott-Parker and S. Zadek. Managing diversity: A key factor in improving efficiency, productivity, and overall business success. *Journal of Vocational Rehabilitation*, 16(2):119–123, 2001.

[14] P. Shenoy and D. S. Tan. Human-aided computing:

utilizing implicit human processing to classify images. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 845–854, 2008.

[15] K. Siorpaes and M. Hepp. Games with a Purpose for the Semantic Web. *IEEE Intelligent Systems*, 23(3):50–60, May/June 2008.

[16] Q. Su, D. Pavlov, J.-H. Chow, and W. C. Baker. Internet-scale collection of human-reviewed data. In *International World Wide Web Conference*, pages 231–240, 2007.

[17] J. Surowiecki. *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*. Doubleday, May 2004.

[18] L. von Ahn. Games with a Purpose. *IEEE Computer*, 39(6):92–94, June 2006.

[19] L. von Ahn and L. Dabbish. Labeling Images with a Computer Game. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 319–326, 2004.

[20] L. von Ahn and L. Dabbish. Designing games with a purpose. *Communications of the ACM*, 51(8):58–67, August 2008.

[21] L. von Ahn, S. Ginosar, M. Kedia, and M. Blum. Improving Image Search with PHETCH. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1209–1212, 2007.

[22] L. von Ahn, S. Ginosar, M. Kedia, R. Liu, and M. Blum. Improving accessibility of the web with a computer game. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 79–82, 2006.

[23] L. von Ahn, M. Kedia, and M. Blum. Verbosity: A Game for Collecting Common-Sense Facts. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 75–78, 2006.

[24] L. von Ahn, R. Liu, and M. Blum. Peekaboom: A Game for Locating Objects in Images. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 55–64, 2006.

[25] D. H. Wilson, A. C. Long, and C. Atkeson. A context-aware recognition survey for data collection using ubiquitous sensors in the home. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1865–1868, 2005.