From Labor to Trader: Opinion Elicitation via Online Crowds as a Market

Caleb Chen Cao † Lei Chen † H. V. Jagadish ‡
†The Hong Kong University of Science and Technology, Hong Kong SAR, PR China
‡University of Michigan, Ann Arbor, MI, USA
{caochen,leichen}@cse.ust.hk, ‡jag@umich.edu

ABSTRACT

We often care about people's degrees of belief about certain events: e.g. causality between an action and the outcomes, odds distribution among the outcome of a horse race and so on. It is well recognized that the best form to elicit opinion from human is probability distribution instead of simple voting, because the form of distribution retains the delicate information that an opinion expresses. In the past, opinion elicitation has relied on experts, who are expensive and not always available. More recently, crowdsourcing has gained prominence as an inexpensive way to get a great deal of human input. However, traditional crowdsourcing has primarily focused on issuing very simple (e.g. binary decision) tasks to the crowd. In this paper, we study how to use crowds for *Opinion Elicitation*. There are three major challenges to eliciting opinion information in the form of probability distributions: *a*) how to measure the quality of distribution; *b*) how to aggregate the distributions; and, *c*) how to strategically implement such a system.

To address these challenges, we design and implement *COPE* (Crowd-powered OPinion Elicitation market). *COPE* models crowdsourced work as a trading market, where the "workers" behave like "traders" to maximize their profit by presenting their opinion. Among the innovative features in this system, we design *COPE updating* to combine the multiple elicited distributions following a Bayesian scheme. Also to provide more flexibility while running *COPE*, we propose a series of efficient algorithms and a *slope* based strategy to manage the ending condition of *COPE*. We then demonstrate the implementation of *COPE* and report experimental results running on real commercial platform to demonstrate the practical value of this system.

Categories and Subject Descriptors

H.2.8 [**DATABASE MANAGEMENT**]: Database Applications—*Data mining*; H.1.2 [**MODELS AND PRINCIPLES**]: User/Machine Systems—*Human information processing*

Keywords

Crowdsourcing; Human Computation; Social Media; Market

KDD'14, August 24–27, 2014, New York, NY, USA.
Copyright 2014 ACM 978-1-4503-2956-9/14/08 ...$15.00.
http://dx.doi.org/10.1145/2623330.2623717.

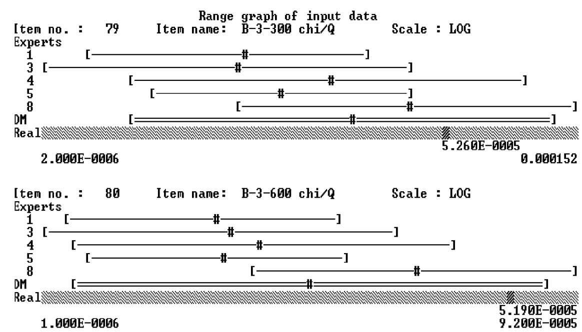


Figure 1: Example of Opinion Elicitation of five participants over two variables(NRC-EU accident uncertainty analysis [4])

1. INTRODUCTION

The application of crowdsourcing has ushered in a brand new age of human-computer collaboration. Online labor markets, such as Amazon MTurk[1], oDesk[2], enable large scale crowdsourcing by providing access to human computation "workers" as well as a programmable infrastructure. The basic assumption in crowdsourcing is that the tasks must be broken down into simple units, such as multiple-choice questions. There is considerable skill in breaking down a complex task into simple units that can be crowd sourced.

Often, we may benefit from getting a richer input from workers. For a very simple example, consider the question 'how many inches of snow will fall tomorrow in city X?' We could discretize this, say into integers, and ask each worker to pick a snipe number as prediction. But such an approach will miss the nuance that the worker predicts 3-6 inches, and chose to predict 4 inches because a single discrete choice was required. For this worker, 6 inches is a likely outcome, while 7 inches is unlikely, but the prediction of 4 inches does not tell us this. In short, simple voting does not suffice because it coarsens the distribution and leads to an extremized opinion expression. What we would like is a framework for eliciting a probability distribution from a worker, and aggregating this with similar distributions obtained from other workers. We call such distributions as *opinions*.

Opinion is a concept of wide extension, which refers to numerical statements expressing an individual's degrees of belief about certain events [17]. Rooted from this definition are such things as prior, posterior, structural distribution, as well as additive probability and belief functions, which can all be expressed in a form of distribution. Such opinion information can then be used not just frivolous pursuits, such as predicting amount of snow, but also as a

[1]www.mturk.com
[2]www.odesk.com

foundation for more advanced decision making procedures and to construct more sophisticated models for risk analysis [4].

In this paper, we focus on exploiting the intrinsic human ability to estimate an opinion, and develop a crowdsourcing framework for this purpose. Specifically, given an event space $\mathcal{S}$, the "workers" are expected to contribute their frank opinion $\vec{r_i}$ in the forms of probability distribution. When the process closes according to certain conditions, the aggregated probability distribution is the elicited consensus opinion.

EXAMPLE 1. *In the standard Probabilistic Risk Analysis [4], during the construction of the Event Trees, one significant step is to provide an accurate probability distribution among many possible outcomes. For certain cases, such probability could be obtained analytically(e.g. failure probability in Circuit Design), but for many sociological or business-related situations, the distributions have to be gathered from a group of related individuals [12]. For example, the Council of Government plans to change the arrangement of public holidays, and before they make the decision, the council board expects a prior distribution among all possible outcomes, e.g. less tourism, more tourism or simply neutral to the alternation. Thus, a crowdsourced application could help to collect the distributed beliefs efficiently.*

EXAMPLE 2. *Another type of applications appears in the causality structure determination, which is widely utilized in building Probabilistic Graphical Models [22]. Given an event, many individuals are invited to describe a set of possible causes for the outcome, along with their probabilistic distributions. Such distribution information can help to build more sophisticated quantitative models such as Bayesian Networks and so on. Figure 1 illustrates an example where 5 expert participants elaborate their probabilistic estimation on two environmental variables B-3-600 and B-3-300 in terms of uniform distribution on selected range over a value space. Then a set of sophisticated inference can be conducted [4].*

There is a crucial challenge for such *Opinion Elicitation* on current crowdsourcing platforms: opinion distribution does not have a "ground truth" with ordinary meaning. Formally, suppose for individual i, the intrinsic opinion for her on an m outcomes event is a probability distribution $\vec{d} = \{d_1, d_2, \ldots, d_m\}$, but during the elicitation process, her reported estimation might be $\vec{r} = \{r_1, r_2, \ldots, r_m\} \neq \vec{d}$. The reason of insincerity might be carelessness or indifference: since the "workers" from online labor markets are essentially money-driven, sloppiness naturally appears when their performance cannot be measured and eventually does not matter. In such cases, we cannot benefit from the correctness based payoff mechanism currently popular on crowdsourcing platforms.

The absence of ground truth knowledge has previously been addressed, most famously in the ESP game [1], where the payoff depends on matching with a partner (or another worker). The basic idea is that if two (or more) individuals independently think of a label for an image, that label is likely to be a good choice. Inappropriate labels are unlikely to qualify. As presented, this is a discrete scheme, there either is a match or there isn't. We would like to extend this idea to find a continuous function to score how well probability distributions match. Furthermore, rather than matching between arbitrarily paired individuals, we do so between each individual and the group 'consensus'.

To address these needs, we propose a 'market' framework for crowdsourcing. Workers trade in this market to earn as high a profit as they can. The concept of variable pay for work is already accepted and widely practiced – many Human Intelligence Tasks (HITs) are offered with a base pay and a (quality-dependent) bonus, with the bonus sometimes being substantially more than the base pay. The quality determination for the bonus is sometimes subjective. We propose instead to have the bonus be determine by the market in a manner that is clearly defined, completely transparent, and reflective of opinion correctness (in a manner we will formally specify below). We implement these ideas in *COPE* (Crowd-powered OPinion Elicitation market). The base pay is provided to workers in the form of seed capital that they invest in an opinion elicitation market with their opinion. We use Bayesian updating, beginning with our initial guess as the prior, to obtain a posterior distribution that reflects the weighted opinions of all the traders in the market. When the task ends, the payoff for each trader is proportional to her contributed modification from the prior to the posterior. Traders that have a zero, or negative contribution, receive no payoff (or a small fixed payoff). In other words, traders have an economic incentive to align their opinion with what they believe will be the consensus direction. Sometimes, individuals may hold opinions that they know are different from the general view. In most opinion elicitation scenarios, we are less interested in such idiosyncratic opinions and more interested in the consensus opinion. The economic incentives of our scheme align well, even for such idiosyncratic individuals, motivating them to express what they truly believe to be popular opinion.

In this paper, we have made the following contributions:

- We formally propose the crowdsourcing application of *Opinion Elicitation*, which treats every crowdsourcing worker as an intrinsic opinion provider and elicits probability distribution information from them.
- We propose *COPE* as the elicitation mechanism, which guarantees the reliability of elicitation and cost control. The *COPE* is built upon a general crowdsourcing worker market, and we strategically induce every involved worker to behave as a risk-neutral trader to achieve honest information elicitation.
- We propose a Bayesian scheme based distribution updating method to incorporate distribution reports from the traders with correctness proof. We also design a series of algorithms to accelerate the incremental merging of causality distribution as well as a *slope* based market running strategy to provide more operational flexibility on *COPE*.
- We practically implement *COPE* on a general labor market and test the system with real datasets. The *COPE* not only serves as a practical information market, but also provides extra human computation resource as a side product.

The rest of the paper is organized as follows. In Section 2 we present a framework overview about the *COPE* system. In Section 3 we formally present the design of *COPE* with necessary preliminaries. In Section 3 we present the algorithmic details about the *COPE* system and a set of core algorithms. In Section 4 we introduce the mechanism of running the system of *COPE*. In Section 5 we elaborate on the non-trivial implementation of *COPE* on Amazon MTurk. In Section 6 we illustrate the experimental results. Then in Section 7 we introduce the recent related work. We conclude the paper with discussion in Section 8.

2. FRAMEWORK OVERVIEW

In this section, we present a system overview of *COPE*(Figure 2). Its three major components are introduced one by one as follows.

Pre-market Building

A running market entails traders and trading capital. In order to turn a crowdsourcing "worker" into an informative "trader", the

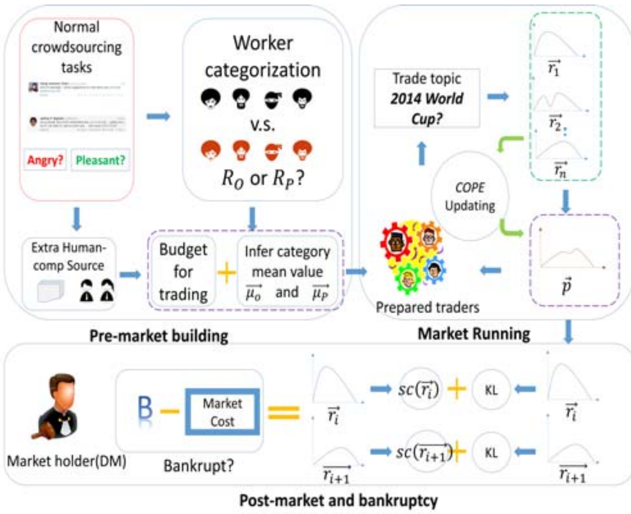


Figure 2: The Framework of *COPE*

Pre-market Building module is designed. First, each "worker" is given a set of normal decision making tasks(termed as *deposit tasks*), e.g. sentiment analysis of short textual messages, image tagging, simple ranking according to image content and so on. After the "worker" finishes these tasks, the promised rewards C_i is retained and set as the *seed capital*. Note that the nature of these tasks will not affect the performance of the market, and these extra crowd-sourced tasks are a by-product from running *COPE*.

Meanwhile, among the normal crowdsourcing tasks, a set of testing tasks, of which we know the true answers, are planted. We then examine a workers responses to these testing tasks to find patterns indicating systematic bias, which helps to calibrate the distribution aggregation afterwards. More details can be found in Section 3.2.2.

Market Running

When the traders are prepared, with their *seed capital* and categorical labels, the *COPE* is initiated. The system presents to each trader a trading topic, i.e. a set of exclusive outcomes of one action or possible reasons for one phenomenon, and each trader may specify a distribution among these options. The trader is informed that their later payoff depends on how much their presented answer contributes to the final global distribution of the entire market. The final payoff will be given when the market is closed by the requester or when other conditions are met like *Market Bankruptcy*. Please refer to Section 3.1.2 for more details about the payment. *COPE* then receives a report $\vec{r_i}$ from each trader t_i, and the *COPE* method is incurred to merge the new reports into the global distribution $\vec{p}$.

Market Supervision

The market holder, a.k.a. *decision maker* monitors *COPE* from the *market supervision* module, where the overall cost of the market is calculated whenever a new updating is conducted. We assume that the decision maker is working under the constraint of a patron budget B. She may choose to end *COPE* when the overall cost exceeds the preset budget. In addition, she is allowed to adopt a *slope* based running strategy to flexibly decide the terminating condition. More details can be found in Section 4.

3. DESIGNING *COPE*

In this section, we first present some basic concepts required to define the information market and probability aggregation. Then we formally describe the design of the *COPE*.

3.1 Preliminaries

3.1.1 *Trader*

The atomic units that build up the *COPE* are the traders. A trader in *COPE*, invests an asset by presenting a report $\vec{r}$ and gets profit by receiving the payoff depending on the presented report $M(\vec{r})$.

Specifically, a *trader* T_i is able to provide a report $\vec{r}$ as a probability distribution over the event space $\mathcal{S}$ of a set of random variables $X = \{X_1, X_2, \ldots, X_m\}$. The report $\vec{r} \in \mathcal{P}$ belongs to the possible report space $\mathcal{P}$, and a trader T_i chooses to present $\vec{r}$ rather than any other possible report $\vec{r'} \in \mathcal{P}$ when the corresponding payoffs $M(\vec{r}) > M(\vec{r'})$. That is, a trader will present a report to maximize his/her payoff according to a certain reward rule.

Without loss of generality, we assume in this work that all *traders* in *COPE* are *Risk-neutral Traders*, which is commonly utilized in Probabilistic Risk Analysis. This property refers to the traders who have no preference between choices with equal expected payoffs(profit), neglecting the risk associated to each option. In other words, a *Risk-neutral Trader* behaves/invests only according to the expected payoffs(profit).

DEFINITION 1 (RISK-NEUTRAL TRADER - T_i). *Given two optional reports $\vec{r}$ and $\vec{r'}$ and their corresponding payoffs $M(\vec{r})$ and $M(\vec{r'})$ as two real-valued random variables, a Risk-neutral Trader T_i considers $\vec{r} \succ \vec{r'}$ iff $E[M(\vec{r})] > E[M(\vec{r'})]$. Here the operation $\succ$ indicates preference among actions.*

For simplicity, in this paper we use simply the term *trader* to refer to a *Risk-neutral Trader*. Note that in the effort of approximating market opinion with the mean beliefs of the traders, risk aversion plays a positive role as well as inevitably introducing certain bias [33]. We consider more detailed tuning on risk-aversion stimulation as the future work, which can be strategically leveraged by adjusting the payoff function in the section below.

3.1.2 *Payoff*

While *COPE* is running, the market maintains a global probability distribution $\vec{p}$, which combines all previous reports from traders. If the *COPE* stops at step n, i.e. n *traders* have finished their investment by providing reports $\vec{R} = \{\vec{r_1}, \vec{r_2}, \ldots, \vec{r_n}\}$, the *COPE* conducts the *payoff* according to their contributed estimated distribution. In the design of *COPE*, we first use KL-divergence to measure the distance between two discrete distributions:

DEFINITION 2 (KULLBACK-LEIBLER DIVERGENCE [23]). *Let $\vec{r}$ and $\vec{u}$ be two probability mass functions in a discrete domain $\mathcal{S}$ with a finite or countably infinite number of values. The Kullback-Leibler diverge(KL divergence) between $\vec{r}$ and $\vec{u}$ is*

$$D(\vec{r}||\vec{u}) = \sum_{x \in \mathcal{S}} \vec{r}[x] \log \frac{\vec{r}[x]}{\vec{u}[x]} \tag{1}$$

KL divergence is defined where for any x in space $\mathcal{S}$ if $\vec{r}[x] > 0$ then $\vec{u}[x] > 0$.

There is a wide spectrum of distance measurements that can serve to gauge the "contribution" from prior distribution $\vec{p}$ to posterior distribution $\vec{p^*}$ by a report $\vec{r_i}$. Considering the scenario of *COPE*, we prefer KL-divergence(also formally known as *relative entropy*) to measure the information gain between two distributions. There are two major reasons: *a*) the process of eliciting causality from crowdsourcing traders is an information gaining process, and KL-divergence is one of the natural practices followed in Bayesian inference or updating [22]; *b*) in accordance to the measurement of the intrinsic *goodness* of a report $\vec{r_i}$(introduced in Section 3.1.3), which is in essence a measure of information entropy, the preference of relative entropy seamlessly incorporates the *COPE Updating* and corresponding *payoff*.

The payoff to a trader T_i considers the relative contribution to the final distribution $\vec{p}$ from her(measured as KL-divergence $D(\vec{r_i}||\vec{p})$). Meanwhile, it is designed to guarantee the honesty of each trader as well. Therefore, we propose the following *payoff* mechanism which incorporates a non-linear component that is related to the final outcome:

DEFINITION 3 (PAYOFF). *Given the market estimation $\vec{p}$, a trader T_i, whose proposed report is $\vec{r_i}$, will receive a payoff M_i, when the market is closed.*

$$M_i = C_i \cdot \frac{Odd}{D_i + 1} = C_i \cdot \frac{Odd}{D(\vec{r_i}||\vec{p}) + 1} \tag{2}$$

C_i is the invested capital of T_i, and Odd is the preset parameter such that at most a trader could earn $Odd \times C_i$ as payoff.

In Definition 3 we use $D(\vec{r_i}||\vec{p})$ instead of $D(\vec{p}||\vec{r_i})$ in order to avoid the zero value in $\vec{r_i}$, which may affect the correctness of the KL-divergence. In addition, to enhance the practical usage, currently we assume the *invested capital* C_i for each trader T_i is the same seed capital: $C_1 = C_2 = \ldots = C_i = \ldots = C$, and the payoff can be thus simplified as

$$M_i = \frac{C \cdot Odd}{D(\vec{r_i}||\vec{p}) + 1}$$

The payoff mechanism in Definition 3 enables *COPE* to punish careless or wrong traders by confining the rewards in the range of $(0, Odd \times C_i]$, where the condition for equality is that the report $\vec{r_i} = \vec{p}$ the global posterior distribution. We summarize and provide the following lemma:

LEMMA 1 (PAYOFF RANGE). *The range of payoff for trader T_i is as follow:*

$$0 < M_i \leq Odd \times C_i \tag{3}$$

The maximum equality is observed when $\vec{r_i} = \vec{p}$.

PROOF. Based on Equation 1, we notice that

$$\begin{aligned}
D(\vec{r_i}||\vec{p}) = & \textstyle\sum_{j \in \mathcal{S}} \vec{r_i}[j] \log \frac{\vec{r_i}[j]}{\vec{p}[j]} && (4)\\
= & -\textstyle\sum_{j \in \mathcal{S}} \vec{r_i}[j] \log \frac{\vec{p}[j]}{\vec{r_i}[j]} && (5)\\
\geq & -\log \textstyle\sum_{j \in \mathcal{S}} \vec{r_i}[j] \frac{\vec{p}[j]}{\vec{r_i}[j]} && (6)\\
= & \log \textstyle\sum_{j \in \mathcal{S}} \vec{p} && (7)\\
= & \ 0 && (8)
\end{aligned}$$

The inequality in line 6 derives from Jensen's Inequality and the convexity of $\log(\cdot)$ function. Thus, we could obtain that

$$0 < \frac{Odd}{D(\vec{r_i}||\vec{p}) + 1} \leq Odd \tag{9}$$

where Lemma 1 is proved. And without difficulty, we can observe that the turning point for a no-loss-no-gain investment is $Odd - D(\vec{r_i}||\vec{p}) = 1$. □

The introduction of KL-divergence as a leveraging parameter in computing the payoff helps the market holder to encourage informative traders. Obviously, this mechanism converts the labor market into a trader market without introducing real-money gambling settings which constrains the development of information markets in a long list of countries [18]. Moreover, currently the design of *COPE* does not allow traders to forsake the opportunity to become a trader, i.e. that must confront the risk of losing the *seed capital*. The mechanism where abandonment of risking will be explored as future work.

Table 1: Summary of Notations

Notation	Description
T_i	a Risk-neutral trader with index i
$\vec{p}$	the current market estimation
$\vec{r_i}$	the report from trader i
$\vec{d_i}$	the subjective belief of trader i
$sc(\cdot)$	the goodness measure function
Odd	the *boosting parameter* set by market holder
$\vec{\mu_o}$	the mean of optimistic distributions
$\vec{\mu_p}$	the mean of pessimistic distributions
$C(C_i)$	the *seed capital* (of trader T_i)
M_i	the *payoff* of trader T_i
λ	the *adjusting parameter* for *COPE* updating

3.1.3 Goodness of Reported Distribution

To reward the traders who contribute reports with more care and honesty, the *COPE* incorporates the factor $sc(\vec{r_i})$ to evaluate the goodness of the contributed report $\vec{r_i}$.

DEFINITION 4 (GOODNESS). *Given a proposed report $\vec{r_i}$ from trader T_i, we define the goodness of the report $sc(\vec{r_i})$as its expected score according to scoring rule in logarithm form [30], i.e.*

$$sc(\vec{r_i}) = \sum_j \vec{r_i}[j] \cdot S_j(\vec{r_i}[j]) = \sum_j \vec{r_i}[j] \log \vec{r_i}[j] \tag{10}$$

The definition of *goodness* $sc(\vec{r_i})$ describes the degree to which the trader tends to present a more extreme distribution(which is more informative) than an even one, which helps the *COPE* adjust the incremental update from each new report $\vec{r_i}$. Specifically, we notice that the value of *goodness* $E(sc)$ is mathematically the entropy of the given report $\vec{r_i}$, which reflects the amount of new information that brought in by trader T_i.

Note that we avoid using a traditional strict scoring rule [18] as payoff mechanism mainly for two reasons: *1*) Strict scoring rules-based mechanisms are mostly suitable for verifiable tasks; and *2*) It is too intricate to explain the sophisticated strict scoring rule to workers from Labor Markets, and even with a explicit explanation, the haste while specifying distribution weakens the rule-oriented calibration to a large extent.

We tentatively present the following conjecture for further exploration:

CONJECTURE 1. *The risk-neutral trader invests honestly under the payoff mechanism in COPE, i.e.*

$$\vec{r_i} = \vec{d_i} \tag{11}$$

Since any linear transformation of existing strict scoring rules is also a strict scoring rule, thus, a light modification of the current payoff mechanism on *COPE* may validate the above conjecture.

3.2 Mechanism of *COPE*

The system of *COPE* is an integrated market that is built on a general crowdsourcing platform. It receives trades (bets) as elicited distribution from traders and combines them into a global distribution. In this section, we present the design and mechanism of three major technique steps that enables the *COPE*.

3.2.1 Evaluating $\vec{r_i}$

Each trader T_i will accomplish several normal crowdsourcing tasks before she/he is entitled to participate into a trading. By injecting probe tasks, whose answers are known beforehand, into normal ones, we could evaluate the bias type of the trader T_i .

After finishing the given batch of normal crowdsourcing tasks, each worker is labeled into one of two categories, namely optimistic $R_o = \{\vec{r^o_1}, \vec{r^o_2}, \ldots, \vec{r^o_{|R_o|}}\}$ and pessimistic $R_p = \{\vec{r^p_1}, \vec{r^p_2},$

Algorithm 1: Overall Procedure while Running *COPE*

Input: Outcome Space $\mathcal{S}$, Market Holder's Prior Distribution $\vec{p_0}$, Budget $B > 0$, n Risk-neutral traders
Output: Posterior Distribution $\vec{p^*}$, payoff M_i to each trader T_i

Initialization: Set the prior distribution $\vec{p_0} = \vec{u}$ as uniform distribution
Pre-market Building: For each trader T_i, label it according to his category, calculate the marginal mean $\vec{\mu_o}$ and $\vec{\mu_p}$ according to Equation 12 and 13
Market Running: For each trader T_i, the Market obtains an discrete distribution $\vec{r_i}$ over the outcome space $\mathcal{S}$
for *each trader* T_i **do**
    adjust the global posterior $\vec{p^*}$ according to ***COPE*** updating;
Post-market:Calculate bounds OR exact Market Cost MC based on Section 4;
if *Bankruptcy conditions are met* **then**
    stop the market;
    conduct payoff to each trader T_i;
return posterior distribution $\vec{p^*}$;

$\ldots, r^{\vec{p}}_{|R_p|}\}$. For each category, an arithmetic mean is calculated and set as the estimation of the first moment of that category.

Specifically, for the category of optimistic workers, we have:

$$\vec{\mu_o} = \frac{\sum_i \vec{r_i^o}}{|R_o|} \tag{12}$$

And for the pessimistic workers, we have:

$$\vec{\mu_p} = \frac{\sum_i \vec{r_i^p}}{|R_p|} \tag{13}$$

Note that to achieve higher accuracy from *COPE*, more categories can be introduced. And the estimated moment values for each category serve as a clue for the following Bayesian scheme distribution updating.

3.2.2 Updating New Report $\vec{r_i}$ into $\vec{p}$

One of the core technique challenges for building the *COPE* is how it should aggregate the elicited reports from individuals into a universal distribution, while following certain guidance or principles that suits the semantic requirements from the market requester or decision maker. This is a controversial topic that incurs long debates among the community of subjective probability [4, 11, 12], and two major types of updating schemes, namely *Axiomatic* and *Bayesian* both exhibit observable rationales under various updating situations. Therefore, the design of updating scheme should be considered together with the specific application scenario. As mentioned in the introduction, the market holder establishes *COPE* mainly in order to elicit a distribution for further risk analysis tasks, which implies the existence of a latent *decision maker* who would conduct an exclusive decision among the possible outcomes. According to C. Genest and J. Zidek's conclusion in [17], the most appropriate method for updating the distribution under such circumstances should follow the Bayesian scheme with a set of specific constraints. In this section, we present the mechanism of *COPE* to update the current global distribution $\vec{p}$ following such a scheme.

In the combining procedure, the current global distribution is treated as the prior distribution, and the aggregated distribution is then treated as the posterior distribution. Following the Bayesian updating scheme, we have following updating expression:

$$p^* = \Pr(\vec{p}|\vec{r}) \propto \frac{\Pr(\vec{p})L(\vec{r}|\vec{p})}{\Pr(\vec{r})} \tag{14}$$

However, in the setting of *COPE* or other running information market, it is practically impossible for the decision maker to assess a full likelihood function $\Pr(\vec{p})L(\vec{r}|\vec{p})$. Fortunately, powered by the normal deposit tasks finished by each trader, we could partially evaluate their tendency with the estimation of the first moment value of each category. Then we are able to devise the *COPE* combining method following standard Bayesian scheme.

First we introduce two essential properties that describe a normative combining method within Bayesian scheme, the *Unanimity Principle* and *Compromise Principle*.

PROPERTY 1 (UNANIMITY PRINCIPLE). *If the reports are the same, the prior distribution should be same to the posterior distribution.*

PROPERTY 2 (COMPROMISE PRINCIPLE). *The posterior distribution should reside in the range of the two extreme reports.*

It is necessary to follow these two principles in order to design a updating method that confirms with Bayesian Updating. The work in [16] studies the expert opinion combination in terms of forecasting tasks, and the proposed "GS-I" model only entails partial assessment of the forecasters to complete the Bayesian updating. Inspired by this, we design the *COPE* Updating as follows.

DEFINITION 5 (*COPE* UPDATING). *For one step updating, the posterior distribution is defined as below:*

$$p^* = p + \lambda(\vec{r_i} - \vec{\mu}) \tag{15}$$

and the λ is the confidence from the decision maker for the extreme values in trader's distribution.

$$\lambda = \frac{sc(\vec{r_i})}{log|\mathcal{S}|} \cdot UB \tag{16}$$

where $|\mathcal{S}|$ is the size of the possible event space and $UB = \min\{\frac{p}{\mu}, \frac{1-p}{1-\mu}\}$ gives the upper bound of the parameter λ.

Here λ is called *adjusting parameter* which helps the decision maker to gauge the mis-calibration or bias of the reporting individuals. Note that since the *goodness* of a report $sc(\vec{r_i})$ is the entropy of the given report, we can obtain the fact that $\lambda \in [0, \min\{\frac{p}{\mu}, \frac{1-p}{1-\mu}\})$ with the following lemma.

LEMMA 2. *The range of the adjusting parameter λ is within* $[0, \min\{\frac{p}{\mu}, \frac{1-p}{1-\mu}\})$.

PROOF. Suppose $U(x) = \frac{1}{|\mathcal{S}|}$ is the probability mass function of a uniform distribution over outcome space $\mathcal{S}$, and $\vec{r_i}$ is the report vector from trader T_i. Then

$$\begin{aligned} D(\vec{r_i}||U) &= \textstyle\sum_{j=1}^{|\mathcal{S}|} \vec{r_i}[j] \log \frac{\vec{r_i}[j]}{U[j]} && (17)\\ &= \log|\mathcal{S}| - H(\vec{r_i}) && (18)\\ &= \log|\mathcal{S}| - sc(\vec{r_i}) && (19)\end{aligned}$$

Then according to proof 1, we know that $D(\vec{r_i}||U) \geq 0$. Thus we have the following inequality.

$$0 \leq D(\vec{r_i}||U) = \log|\mathcal{S}| - sc(\vec{r_i}) \tag{20}$$

which indicates that

$$0 \leq \lambda = \frac{sc(\vec{r_i})}{log|\mathcal{S}|} \cdot UB \leq UB \tag{21}$$

and Lemma 2 is proved. $\square$

Algorithm 2: Procedure of Computing Market Cost of *COPE*

Input: current global posterior distribution $\vec{p}$, reports from n traders $R_n = \{\vec{r_1}, \vec{r_2}, \dots, \vec{r_n}\}$, *seed capital* C, preset *boosting parameter Odd*

Output: total Market Cost MC

Initialization: zero-valued vector $\vec{M}$ with size n, and $MC = 0$

for *each* $\vec{r_i}$ *in* R_n **do**

  compute M_i according to Equation 2 ;

  $\vec{M}[i] = M_i$;

  $MC+ = \vec{M}[i]$;

return MC;

Then the benefit of introducing *COPE* Updating method can be observed: the design of *COPE* naturally satisfies the aforementioned two basic principles. We then claim the following lemma.

LEMMA 3 (PROPERTIES OF *COPE*). *The COPE updating confirms with the Unanimity Principle and Compromise Principle.*

PROOF. According to Theorem 2.1 in [16], an updating strategy satisfies the two properties when the value of λ observes the following condition:

$$\max\{\frac{p}{\mu - 1}, \frac{p-1}{\mu}\} \leq \lambda \leq \min\{\frac{p}{\mu}, \frac{1-p}{1-\mu}\}. \tag{22}$$

Since $\lambda = \frac{sc(\vec{r_i})}{log|\mathcal{S}|} \in (0, 1)$, it satisfies the requirement above. Thus the two properties are satisfied. □

3.3 Cost of *COPE*

To establish a *COPE*, the market holder has to be willing to be a patron with a budget B: when the overall cost of the *COPE* exceeds this budget, the market has to be closed(termed as *bankruptcy*). Of course, the patron could close the market before this, if a sufficiently good result has been obtained.

DEFINITION 6 (MARKET COST). *Given a COPE, the current distribution $\vec{p}$, the set of reports from n traders $R_n = \{\vec{r_1}, \vec{r_2}, \dots, \vec{r_n}\}$, the Market Cost is defined as the summation of the payoffs from every trader t_i, i.e.*

$$MC = MC(\vec{p}, R_n) = \sum_{i=1}^{n} M_i \tag{23}$$

The ending condition for one instance of *COPE* is that MC exceeds the preset budget B. And when this condition is reached, the market stops accepting any new report and returns to the market holder with the current global distribution $\vec{p^*}$. This condition has to be checked after every trade report, and can be expensive to compute. In Sec. 4.3 we describe how to do this efficiently.

4. RUNNING *COPE*

In this section, we introduce the essential techniques to keep the *COPE* running. First we introduce the algorithm conducting the *COPE* Updating Strategy in a batching style. Then we introduce the computation of the *Market Cost* given the set of reports $R_n = \{\vec{r_1}, \vec{r_2}, \dots, \vec{r_n}\}$. Moreover, we further introduce an advanced strategy to run the market with more flexibility. An overall procedure to run the *COPE* is presented in Algorithm 1.

4.1 Batch *COPE* Updating

In Section 3.2.2, we introduce the mechanism of updating the current global distribution $\vec{p}$ from one individual report $\vec{r_i}$. For each iteration of such individual update, the algorithm takes $\mathcal{O}(|\mathcal{S}|)$ time complexity. If given an entire set of reports $R_n = \{\vec{r_1}, \vec{r_2}, \dots, \vec{r_n}\}$, the total time complexity becomes $\mathcal{O}(|\mathcal{S}| \cdot n)$. In this section, we introduce a close form of conducting *COPE* updating in batch.

THEOREM 1 (BATCH *COPE* UPDATING). *Given n individual reports $R_n = \{\vec{r_1}, \vec{r_2}, \dots, \vec{r_n}\}$, and the labeled category mean $\vec{\mu_o}$ and $\vec{\mu_p}$, the posterior global distribution over outcome set $\mathcal{S}$ can be computed following the equations below [16]:*

$$\vec{p^*} = \frac{p^{1-n} \prod_{i=1}^{n} n_i}{p^{1-n} \prod_{i=1}^{n} n_i + (1-p)^{1-n} \prod_{i=1}^{n} (1-n_i)} \tag{24}$$

where for each $1 \leq i \leq n, n_i = [p + \lambda_i(\vec{r_i} - \vec{\mu})]$ is between the upper and lower bounds in Equation 22 and $\vec{\mu}$ is to be replaced according to the label of each report.

By conducting the batch-style *COPE* Updating, the time complexity could be reduced to $\mathcal{O}(n + |\mathcal{S}|)$.

4.2 Computing Market Cost

The computation of Market Cost MC is another important issue to consider in order to efficiently manage the *COPE*. After obtaining the posterior global distribution $\vec{p}$ by conducting *COPE* Updating, the market goes through all the reports and calculate the exact payoff for each trader. A sketchy procedure of computing the Market Cost MC is given in Algorithm 2. The time cost of exactly calculating a market cost is $\mathcal{O}(n \cdot |S|)$.

Note that the payoff for each trader is positive, but there are traders who suffer from payoff less than the *seed capital* C due to large difference between their report $\vec{r_i}$ and $\vec{p}$.

4.3 Market Running Strategy

When a market holder runs a *COPE*, he acts both as a patron to support the market and a decision maker to determine when to stop the market. The most simple strategy of running the market is to calculate the exact total market cost every time when a new report comes, and shut down the market immediately when $MC \geq B$, which we term as *Bankruptcy*.

Besides the trivial strategy described above, in this section, we propose a ***slope* based running strategy** which grants the market holder more flexibility, as well as improving the efficiency. The *slope* is formed by the upper and lower edges of the market cost: when the upper edge reaches the preset budget B, the system triggers a warning that the Budget is running out and starts to calculate the exact MC; if the lower edges exceeds the Budget, the market is terminated immediately. But while the Budget is running within the *slope* range, the system keeps the market running and returns to the market holder the current result and wait for the decision of the holder.

To facilitate such a *slope* based strategy, we develop a pair of lower and upper bounds of the KL-divergence based on a variant of Pinsker Inequality.

Lower Slope Edge

First we define the concept of *Variational Distance*, a.k.a. $\mathcal{L}_1$-distance, which can be computed faster than the KL-divergence due to the avoidance of computing weighted logarithm value.

DEFINITION 7 (VARIATIONAL DISTANCE). *Let $\vec{v}$ and $\vec{u}$ be two probability mass functions in a discrete domain $\mathcal{S}$ with a finite or countably infinite number of values. The Variational Distance*

Algorithm 3: Procedure of determining Market Bankruptcy of *COPE*

Input: current global posterior distribution $\vec{p}$, reports from n traders $R_n = \{\vec{r_1}, \vec{r_2}, \ldots, \vec{r_n}\}$, *seed capital* C, preset *boosting parameter* Odd

Output: Decision of *Bankruptcy*

Initialization: *Bankruptcy* bool flag $F_B K = 0$,
for *each* $\vec{r_i}$ *in* R_n **do**
  for *each* j *in* $\vec{r_i}$ **do**
    compute the $V_i(\vec{r_i}, \vec{p}) += |\vec{r_i}[j] - \vec{p}[j]|$;
    compute the $c_i = \max_j(\vec{r_i}[j]/\vec{p}[j])$;
  $M_i^L = Odd \cdot C/(\log 2 \cdot V(\vec{r_i}, \vec{p}) + \log c_i + 1)$;
  $MC^L += M_i^E$;
if $MC^L - B > 0$ **then**
  return $F_B K = 1$(*Bankruptcy*);
else
  return $F_B K = 0$(still running);

Algorithm 4: Procedure of Warning Trigger of *COPE*

Input: current global posterior distribution $\vec{p}$, reports from n traders $R_n = \{\vec{r_1}, \vec{r_2}, \ldots, \vec{r_n}\}$, *seed capital* C, preset *boosting parameter* Odd

Output: Decision of Warning Trigger

Initialization: *Warning* bool flag $F_W N = 0$,
for *each* $\vec{r_i}$ *in* R_n **do**
  //compute Variational Distance $V_i = V(\vec{r_i}, \vec{p})$ according to Equation 25 ;
  for *each* j *in* $\vec{r_i}$ **do**
    $V_i(\vec{r_i}, \vec{p}) += |\vec{r_i}[j] - \vec{p}[j]|$;
  $M_i^U = Odd \cdot C/(\frac{1}{2}V_i^2 + 1)$;
  $MC^U += M_i^U$;
if $MC^U - B > 0$ **then**
  return $F_W N = 1$(*Warning*);
else
  return $F_W N = 0$(still running);

between $\vec{v}$ and $\vec{u}$ is

$$V(\vec{v}, \vec{u}) = \sum_{j=1}^{n} |\vec{v}[j] - \vec{u}[j]| \tag{25}$$

whose value is always non-negative.

We now present the following *Lower Bounding* theorem.

THEOREM 2 (LOWER BOUNDING). *Given a report $\vec{r_i}$, the current global distribution $\vec{p}$, the lower bound of the payoff to trader T_i can be achieved as follow:*

$$M_i^L = Odd \cdot C/(\log 2 \cdot V(\vec{r_i}, \vec{p}) + \log c + 1) \leq M_i \tag{26}$$

where $c = \max_j(\vec{r_i}[j]/\vec{p}[j])$.

Before we prove the given lower bound, we first define the concept of the *Capacitory Discrimination* as follows:

DEFINITION 8 (CAPACITORY DISCRIMINATION). *Let $\vec{v}$ and $\vec{u}$ be two probability mass functions in a discrete domain $\mathcal{S}$ with a finite or countably infinite number of values. The Capacitory Discrimination between $\vec{v}$ and $\vec{u}$ is*

$$C(\vec{v}, \vec{u}) = D(\vec{v}||\vec{m}) + D(\vec{u}||\vec{m}) \tag{27}$$

where

$$\vec{m} = \frac{1}{2}(\vec{v} + \vec{u}) \tag{28}$$

is the arithmetic mean of the two given vectors.

And we introduce the *Pinsker's Inequality* as follows:

THEOREM 3 (PINSKER'S INEQUALITY). *Given two probability mass functions $\vec{v}$ and $\vec{u}$ with outcome space $\mathcal{S}$, the following inequality is observed:*

$$D(\vec{v}||\vec{u}) \geq \frac{1}{2}V^2(\vec{v}, \vec{u}) \tag{29}$$

The equality holds when $\vec{u} = \vec{v}$.

PROOF. Please refer to [13]. □

Then we prove Theorem 2 based on *Variational Distance* and the *Pinsker's Inequality*:

PROOF (THEOREM 2). According to Theorem 4 in [29], we can obtain that

$$D(\vec{r_i}||\vec{p}) \leq C(\vec{r_i}, \vec{p}) + \log(\frac{1}{2}(1 + c)) \tag{30}$$

where $C(P, Q)$ is the *capacitory discrimination* with $\vec{m} = \frac{1}{2}(\vec{r_i} + \vec{p})$

$$C(\vec{r_i}, \vec{p}) = D(\vec{r_i}||\vec{m}) + D(\vec{p}||\vec{m}) \tag{31}$$

based on the definition of Variational Distance in Equation 25 and Pinsker Equality in Equation 29, we could obtain an upper bound of KL-divergence:

$$D(\vec{r_i}||\vec{p}) \leq \log 2 \cdot V(\vec{r_i}, \vec{p}) + \log c \tag{32}$$

which leads to the inequality in the Theorem 2. □

The lower bounds in Theorem 2 enables the *COPE* with a fast algorithm to determine the *Bankruptcy* situation. We elaborate the procedure in Algorithm 3.

Upper Slope Edge

When the upper edge of the slope reaches the preset budget, the market holder receives an warning, and since then the system starts to calculate the exact market cost.

As a warning function, the upper slope edge is realized based on an upper bound of the KL-divergence:

THEOREM 4 (UPPER BOUNDING). *Given a report $\vec{r_i}$, the current global distribution $\vec{p}$, the upper bound of the payoff to trader T_i can be achieved as follow:*

$$M_i^U = Odd \cdot C/(\frac{1}{2}V^2(\vec{r_i}, \vec{p}) + 1) \geq M_i \tag{33}$$

The proof can be directly inferred from the Pinsker Inequality. Equipped with the upper bound of individual payoff, we are able to estimate the upper edge of the slope. We then present the Warning Trigger technique as described in Algorithm 4. Note that while calculating the lower edge, the variational distance $V(p, r)$ has already been calculated, therefore, with a limited extra space cost, we could obtain the upper slope edge with an $\mathcal{O}(1)$ time cost.

5. IMPLEMENTATION

In this section, we illustrate the non-trivial parts of the implementation of *COPE*.

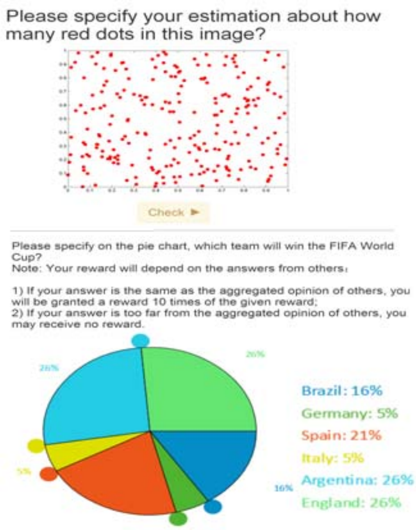


Figure 3: HIT Interface Prototype in a Running *COPE*

5.1 Deposit Task

There are two factors to consider while designing the deposit tasks: the tasks should be with general purpose, and the estimation tendency should be measured for the later Bayesian Updating. Following this requirement, we build up a prototype of *COPE* based on classic jelly-beans-in-a-jar experiment [20]. In this prototype implementation, when a "worker" accepts the task, she is presented with a sample image and informed that the given image contains exactly 200 blue dots. Then another figure is given as shown in the upper part of Figure 3 and the "worker" is required to estimate the number of dots without exhaustively counting them. A set of such tasks could be included in the *pre-market* stage, with a moderate amount of rewards, which makes the "worker" begrudged to forsake. Besides the family of jelly-beans-in-a-jar experiments, many other suitable tasks could be incorporated into the *pre-market* stage. One of the major requirement is that the estimation tendency label of a worker $\vec{r_i}$ should be able to be inferred. As in the prototype, the average over-estimation or under-estimation of the number of dots serves as a measure of the estimation tendency label.

5.2 Opinion Elicitation

After the input from the "worker", another page is presented as shown in the lower part of Figure 3, which is previously hidden. In this new page, the "worker" is required to specify the opinion distribution among a set of options. Due to the "probability-phobia", i.e. common workers feel unwilling or uncomfortable to exactly specify a value of a probability, we use dynamic chart to elicit the probabilistic estimation from a trader. The "worker" is given a set of options where the distribution information is unknown, and she is able to express opinion by swirling the spline of the pie chart. The "worker" is now informed that the final reward depends on how similar her estimation is compared to the aggregated opinion. Under such case, the "worker" is forced into a psychological situation of a "trader", where she confronts a situation of losing certain stake if her answer is not carefully presented. The proposed mechanism in *COPE* is thus activated.

5.3 Payoff Dispatch

The essential technique challenge to implement a *COPE* on general crowdsourcing platform is that the current infrastructure does not support a flexible payment mechanism, which hinters the application of the market. To tackle this challenge, besides the virtual "stake" from the *seed capital*, in *COPE* we also develop an innovative approach to conduct the payoff dispatching. When a "worker" finishes the opinion estimation, a specific task is generated after the closing of the *COPE*, and the task is visible and acceptable only to the "worker" by identifying her worker ID. The reward of the new task is set as the calculated payoff M_i.

6. EXPERIMENTS

In this section, we present the experimental study of the *COPE*. Specifically, we first study the effectiveness of introducing market mechanism in opinion elicitation. We compare the market mechanism with simple voting strategies. After that, we evaluate the relationship between the latency and the number of aggregated traders' information, then we study the characteristic between the distribution properties and the total *Market Cost*. In all figures, we use *ex* to denote the performance of the exact algorithm and *lw* and *up* to denote that of lower and upper bounding algorithms respectively.

6.1 Merits of Market Mechanism

In this section, we compare between the market-based mechanism with simple voting to show the merit of introducing the new mechanism. Note that in a jelly-beans-in-a-jar experiment, the mutual communication among participants impairs the crowd's accuracy [32]. However in the setting of *COPE*, the "investors" only know the existence of other participants but are unaware of others' opinion. Specifically, we set up three set of experiments on Amazon MTurk as follows:

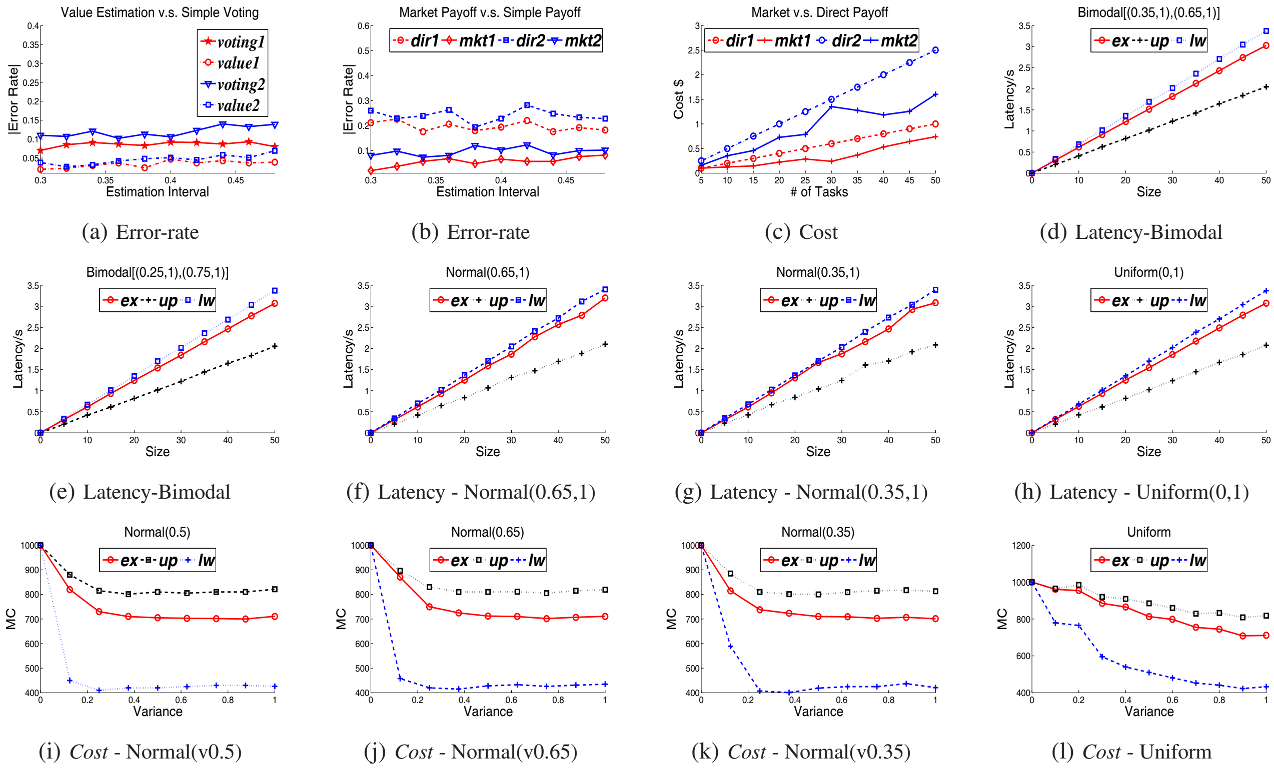
a) We propose to the online crowds a picture with a man's portrait on it, and crowds are asked to estimate the age of the man(ground truth is known to be 40). In the simple voting, we set up two pivots value, e.g. $piv_1 = 35$ and $piv_2 = 45$, and ask the crowds to vote on the option that she think is closest to the real age. Then an answer is achieved by calculating the weighted aggregation. On the contrary, for the value based experiments, crowds are asked to specify a value between piv_1 and piv_2 and their mean is aggregated as the answer. The results are shown in Figure 4(a), where the error rates from simple voting (both $vt1$ and $vt2$) are higher than those of the value estimation ($vl1$ and $vl2$). The experiments are conducted on two independent data sets.

b) We then evaluate the error rates when the crowds are informed with different payoff mechanism, all under value estimation method. The results are shown in Figure 4(b). A first group of crowds are informed that their payoffs are irrelevant to their answers('dir1' and 'dir2'). The last group is encouraged to estimate according to the majority's opinion, i.e. the market mechanism introduced in this paper('mkt1' and 'mkt2'). The results show that the direct payoff incurs highest error rate and the market based method performs best in most cases.

c) Then we test the cost of introducing the market based mechanism. Varying the number of tasks, we record the total payoffs and show the results in Figure 4(c). We set up the price for a single task as 0.02 and 0.05USD respectively. Then the simple direct payoffs are shown in dashed lines and the market based mechanism in solid lines, which saves more payoff for the requester.

6.2 Latency v.s. Market Size

In this and next sections, we mainly study the performance of the proposed algorithms in mainly two aspects: the efficiency according to the number of collected reports, and the value of *Market Cost* according to a varying characteristic of the collected reports. All experiments are conducted on a PC with 2 Intel(R) Core(TM) 2.13GHz CPU and 4GB memory, running on Microsoft 64-bit Win-



(a) Error-rate (b) Error-rate (c) Cost (d) Latency-Bimodal

(e) Latency-Bimodal (f) Latency - Normal(0.65,1) (g) Latency - Normal(0.35,1) (h) Latency - Uniform(0,1)

(i) *Cost* - Normal(v0.5) (j) *Cost* - Normal(v0.65) (k) *Cost* - Normal(v0.35) (l) *Cost* - Uniform

Figure 4: Experiments Results

dows 8. We generate the data sets following both the family of normal distribution and uniform distribution.

We first present the performance of calculating the *Market Cost* with a varying report number from Figure 4(d) to Figure 4(h). Specifically, we test the performance on Bimodal distribution, Normal distribution and Uniform distribution. As described in Section 4, the time complexity of calculating the *Market Cost* is $\mathcal{O}(n \cdot |\mathcal{S}|)$, thus the latency is roughly linear according to a fixed size of outcome space and an increasing number of reports.

6.3 Market Cost v.s. Distribution Variance

In this experiment, we test the relationship between the *Market Cost* and the variance within a reported distribution. Specifically, we simulate the statistical feature of the crowd with the following four distribution: normal distribution with mean value equal to 0.5 (1), normal distribution with mean value equal to 0.65 (2), normal distribution with mean value equal to 0.35 (3), and uniform distribution (4). For each distribution, the market cost is calculated at every variance value ranging from 0 to its set upper limit.

The results are shown from Figure 4(i) to Figure 4(l). From the tested results, we can clearly observe that, for all the distribution, the *Market Cost* is maximized where the variance equals to 0. Then, the *Market Cost* goes down sharply with the variance increasing. In the final stage, the market cost becomes stable. Such a result is easy to interpret: when the variance of the crowd is small, that people hold similar attitudes toward the given event. Therefore, their reports are likely to resemble the market estimation, which makes the market cost large at first. When the variance gets larger, the crowds' opinions become diversified. Then, we can infer that traders' reports are likely to vary greatly from market estimation, which leads to a lower *Market Cost*. The result here also indicates that the variance of the crowd can serve as a good predictor of the market cost.

7. RELATED WORK

In this section, we summarize the literature of research related to the design of *COPE*. As described in Section 1, the design of *COPE* is a pioneering effort to realize a market-manner information aggregator onto a concrete commercial platform.

7.1 Information Elicitation

Probability distribution elicitation has paved its way in two main branches, Scoring Rules [6] and Information Market [19].

Scoring Rules are first adopted as an incentive mechanism to reward weather forecasters [6], where the reward depends on the designed rules and the final outcomes of the uncertain events. Many different scoring rules are then developed to suit various elicitation scenarios: quadratic strict rules [6], logarithm rules [28], and more recently scoring rule for more complex enviroments [8]. In the work [27] by L. Savage, a functional characterization is presented, and a comprehensive study of Scoring Rule to evaluate probability can be found in work [30] by R. Winkler.

Information Market is developed based on the observation that speculative markets like financial markets or commodity markets reflect the confidence of the investors faithfully. Then chances to reduce information uncertainty are wrapped as tradable products on Information Market, where the stable price reflects the information estimation among the investors. Early prototypes can be found in horse betting [19], political polls [5], science progress [25] and so on. A recent effort [18] by R. Hanson et al. proposes to combine the Information Market and Scoring Rule to facilitate a sequential market. Such a Scoring Market Rule is essentially a cost-function market maker [10] from the perspective of market design. This thread of approach also produces new practice of crowdsourced learning tasks [31], where an iterative and interactive learning procedure is proposed to enhance the prediction performance based on given historical dataset(on the contrary, *COPE* addresses the task of elicit intrinsic opinion of workers without any historical training efforts). Moreover, a recent series of work [9, 26] conducted by Y. Chen et al. provides more sophisticated market mechanism for information elicitation. In the work [9] the authors design a special type of information market that capture the signal structure as informative securities. However, the design of such advanced information markets entails a specially designed speculative market, which is way too complicated and intractable on general labor markets. And in the work [26] the authors implement an information polling tool based on Amazon MTurk by direct polling and payoff, without the support of a market structure.

7.2 Data-driven Human Computation

Human computation is a long-existing concept and has been practiced for centuries. Specifically, whenever a "human" serves to "compute", a human computation is observed. This leads to a history of Human Computation even longer than that of electronic computer. However, with the emergence of Internet web service, especially the one that facilitates online labor recruiting and managing like Amazon MTurk and oDesk, human computation starts to experience a new age where the source of human is broadened to a vast pool of crowds, instead of designated experts or employees. This type of outsourcing to crowds, i.e. *crowdsourcing*, is now receiving countless success in many areas such as fund raising, logistics, monitoring and so on.

In data-driven applications, human cognitive abilities are mainly exploited in two types: voting among many options, and providing contents according to certain requirements. The wide usage of "voting" as a human computing action grows from the observation that humans are better at comparisons rather than evaluation objects by specifying an exact numerical value. Most of basic queries in database [14] and data mining [2] can be decomposed into simple voting as human tasks [7, 14]. Meanwhile, in order to break the *close world assumption* in traditional database, human are enrolled to provide extraneous information to answer certain queries: item enumeration [14], content composing [3, 21], counting [24] and so on. The work in this paper takes effort to explore new usage of human cognitive ability, specifying probability estimation among uncertain events.

8. CONCLUSION

In this paper, we design and propose the *COPE* to facilitate *Opinion Elicitation* functions on general crowdsourcing labor market. We also elaborate the rationale of the design in terms of distribution combination and payoff mechanism. In the end, we theoretically and practically testify the performance of such function.

More general Opinion Elicitation tasks observe multivariate random relationship, whose outcome space may increase exponentially, which render the Market unmanageable. However, human cognitive knowledge is able to help decide the necessity of such dependency, while providing quantitative causality information. We will proceed to extend *COPE* to capture such potential in the future.

Acknowledgment

We thank the anonymous reviewers and area chairs for their in-depth review and constructive comments.

This work is supported in part by the Hong Kong RGC Project N_HKUST637/13, National Grand Fundamental Research 973 Program of China under Grant 2012-CB316200, National Natural Science Foundation of China (NSFC) Grant No. 61232018, NSF grant IIS-1250880, Microsoft Research Asia Gift Grant and Google Faculty Award 2013.

9. REFERENCES

[1] L. von Ahn and L. Dabbish. Labeling images with a computer game. In *CHI Conference*,2004.

[2] Y. Amsterdamer, Y. Grossman, T. Milo, and P. Senellart. Crowd mining. In *SIGMOD*,2013.

[3] Y. Baba and H. Kashima. Statistical quality estimation for general crowdsourcing tasks. In *SIGKDD Conference*, 2013.

[4] T. Bedford and R. Cooke. *Probabilistic risk analysis: foundations and methods*. Cambridge University Press, 2001.

[5] J. Berg, F. Nelson, and T. Rietz. Accuracy and forecast standard error of prediction markets. *Tippie College of Business Administration, University of Iowa*, 2003.

[6] G. Brier. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

[7] C. C. Cao, J. She, Y. Tong, and L. Chen. Whom to ask? jury selection for decision making tasks on micro-blog services. *VLDB*, 2012.

[8] Y. Chen and I. Kash. Information elicitation for decision making. *AAMAS*, 2011.

[9] Y. Chen, M. Ruberry, and J.W. Vaughan. Designing informative securities. In *UAI*, 2012.

[10] Y. Chen, M. Ruberry, and J.W.Vaughan. Cost function market makers for measurable spaces. In *ACM EC*, 2013.

[11] R. T. Clemen and R. L. Winkler. Unanimity and compromise among probability forecasters. *Management Science*, 36(7):767–779, 1990.

[12] R. T. Clemen and R. L. Winkler. Combining probability distributions from experts in risk analysis. *Risk analysis*, 19(2):187–203, 1999.

[13] T. M Cover and J. A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[14] M. J. Franklin, D. Kossmann, T. Kraska, S. Ramesh, and R. Xin. Crowddb: answering queries with crowdsourcing. In *SIGMOD*, 2011.

[15] J. Gao, X. Liu, B. C. Ooi, H. Wang, and G. Chen. An online cost sensitive decision-making method in crowdsourcing systems. In *SIGMOD*, 2013.

[16] C. Genest and M. J Schervish. Modeling expert judgments for bayesian updating. *The Annals of Statistics*, 13(3):1198–1212, 1985.

[17] C. Genest and J. V Zidek. Combining probability distributions: A critique and an annotated bibliography. *Statistical Science*, 1986.

[18] R. Hanson. Combinatorial information market design. *Information Systems Frontiers*, 5(1):107–119, 2003.

[19] D. B. Hausch, V. SY Lo, and W. T Ziemba. Efficiency of Racetrack Betting Markets. *Academic Press New York*, 1994.

[20] J. Howe. Crowdsourcing: Why the power of the crowd is driving the future of business. *Crown Publishing Group*, 2008.

[21] L. Kazemi and C. Shahabi. Geocrowd: enabling query answering with spatial crowdsourcing. In *ACM GIS Conference*, 2012.

[22] D. Kollar and N. Friedman. Probabilistic graphical models: principles and techniques. *The MIT Press*, 2009.

[23] S. Kullback and R. A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[24] A. Marcus, D. Karger, S. Madden, R. Miller, and S. Oh. Counting with the crowd. *VLDB*, 2013.

[25] D. M Pennock, S. Lawrence, C L. Giles, F. A. Nielsen, et al. The real power of artificial markets. *Science*, 291(5506):987–988, 2001.

[26] T. Pfeiffer, X. Gao, Y. Chen, A. Mao, and D. G Rand. Adaptive polling for information aggregation. In *AAAI*, 2012.

[27] L. J Savage. Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association*, 1971.

[28] E. H Shuford Jr, A. Albert, and H E. Massengill. Admissible probability measurement procedures. *Psychometrika*, 1966.

[29] F. Topsoe. Some inequalities for information divergence and related measures of discrimination. *Trans. Inf. Theory*, 2000.

[30] R. Winkler, J. Muñoz, José Cervera, José Bernardo, G. Blattenberger, J. Kadane, D. Lindley, A. Murphy, R. Oliver, and D. Ríos-Insua. Scoring rules and the evaluation of probabilities. *JASA*, 1996.

[31] J. Abernethy and R. M. Frongillo. A Collaborative Mechanism for Crowdsourcing Prediction Problems. In *NIPS*, 2011.

[32] J. Lorenz, H. Rauhut, F. Schweitzer and D. Helbing. How social influence can undermine the wisdom of crowd effect. In *PNAS*, 2011.

[33] J. Wolfers and E. Zitzewitz. Interpreting prediction market prices as probabilities. In *NBER Working Paper 12200*, 2006.