Chen et al. *EPJ Data Science* (2016) 5:20
DOI 10.1140/epjds/s13688-016-0082-4

EPJ Data Science
a SpringerOpen Journal

REGULAR ARTICLE Open Access

Bandit strategies in social search: the case of the DARPA red balloon challenge

Haohui Chen[1,2*], Iyad Rahwan[3] and Manuel Cebrian[1]

*Correspondence:
CaronHaohui.Chen@data61.csiro.au
[1]Data61 Unit, Commonwealth Scientific and Industrial Research Organization, Melbourne, Victoria, Australia
[2]Faculty of Information Technology, Monash University, Caulfield, Victoria, Australia
Full list of author information is available at the end of the article

Abstract

Collective search for people and information has tremendously benefited from emerging communication technologies that leverage the wisdom of the crowds, and has been increasingly influential in solving time-critical tasks such as the DARPA Network Challenge (DNC, also known as the Red Balloon Challenge). However, while collective search often invests significant resources in encouraging the crowd to contribute new information, the effort invested in verifying this information is comparable, yet often neglected in crowdsourcing models. This paper studies how the exploration-verification trade-off displayed by the teams modulated their success in the DNC, as teams had limited human resources that they had to divide between recruitment (exploration) and verification (exploitation). Our analysis suggests that team performance in the DNC can be modelled as a modified multi-armed bandit (MAB) problem, where information arrives to the team originating from sources of different levels of veracity that need to be assessed in real time. We use these insights to build a data-driven agent-based model, based on the DNC's data, to simulate team performance. The simulation results match the observed teams' behavior and demonstrate how to achieve the best balance between exploration and exploitation for general time-critical collective search tasks.

Keywords: crowdsourcing; exploration; exploitation; misinformation; disinformation; social search; bandit problem

1 Introduction

Crowdsourcing, the use of the Internet to solicit contributions from large groups of people, has been shown to be very effective in time-critical tasks, ranging from manhunts [1–3], to influenza detection [4], to crisis-mapping [3, 5, 6]. However, time-critical crowdsourcing tasks often reward the collection of new information, but ignore the efforts of verification. Crowds tend to explore new information but seldom verify it autonomously, and exploration effort often dominates. This causes information overload, where misinformation (caused by error) and disinformation (caused by deliberate malice) conceal true information [7], posing a significant challenge to crowdsourcing. In the context of disaster response, while online social media is a highly-effective crowdsourcing tool, it also makes it nearly costless to spread false information [8]. Misinformation has impeded search and rescue operations [5], and sometimes it can go as far as harming innocent people. For example, during the manhunt for the Boston Marathon bombers, the crowd wrongly identified one missing student, Sunil Tripathi, as a suspect. It subsequently emerged that he

had died days before the bombings, yet misinformation was spread in 29,416 tweets [9]. Scholars have identified this problem and paid attention to the detection of false information. Gupta et al. [8] and Boididou et al. [10] building up on Canini et al. [11]'s work, use contents and corresponding authors' profiles to classify false information, and achieve relatively high accuracy. However, in reality, information arrives from various channels, e.g. phone calls, text messages or online social media. Therefore, there is no universal method of processing the information and even classifying it in a short period of time. This paper does not attempt to build a classifier or a universal strategy for discriminating misinformation or disinformation from correct entries. Rather, we assume that, based on the discussion above, the success of a time-critical task requires not just exploring new information (exploration) but also verification (exploitation). Given that an individual or organization has limited resources, exploration and exploitation are regarded as two competing processes [12]. Therefore, this paper explores how to balance exploration and exploitation in time-critical crowdsourcing tasks.

We use DARPA Network Challenge (DNC) as the study case. In 2009, DARPA launched a competition, which aims to evaluate the power of social networks and media in mobilizing crowds. Ten red weather balloons were placed at undisclosed locations throughout the United States. Participating teams or personnel competed to be the first one to locate all the red balloons and win a prize of $40,000 [13]. This paper revisits the full submission history of individual teams, and statistically analyses why high-ranking teams topped the challenge. We found that a large number of false locations were submitted across the teams. Moreover, some of the false locations were submitted concurrently by more than one team, which implies that some teams were using similar sources of information or, as stated by Smith [14], attacks were organized during the competition. As the veracity of sources strongly influences the quality of information [3], to succeed in the DNC, a team must strike a balance between exploring new sources and exploiting the most reliable ones. We assume that the DNC can be modelled as a Multi-Armed Bandit (MAB) problem [15, 16], which implies that solutions for other MAB problems could also be effective. Employing empirical studies of MAB problems, we develop agent-based simulation models to study performance of alternative strategies and to assess the optimal one.

2 Statistics

There were a total of 42 registered teams in the competition. In addition, there was a large amount of anonymous submissions. We assign all anonymous submissions to a virtual team called Anonymous (the complete submission history can be found in [17] and a video illustrating submissions over time is shown in Additional file 1). A submission may contain up to ten locations, and each location comprises a pair of longitude and latitude for a purported balloon. Every team could submit multiple entries while waiting for the previous ones to be validated by DARPA. For each submission DARPA returned the number of correct locations that submission contained. However, the mechanism used by DARPA to screen submissions is not known, neither is the time needed to validate each submission.

When entries were identical in terms of the longitude and latitude, they were treated as copies of the same location. Therefore, we use one mile as the maximum uncertainty in each dimension and locations that overlap within this margin of error are grouped and called a cluster. In addition to the 10 correct clusters (corresponding to 102 correct locations), we found a staggering set of 263 false clusters (constructed by 356 false locations).

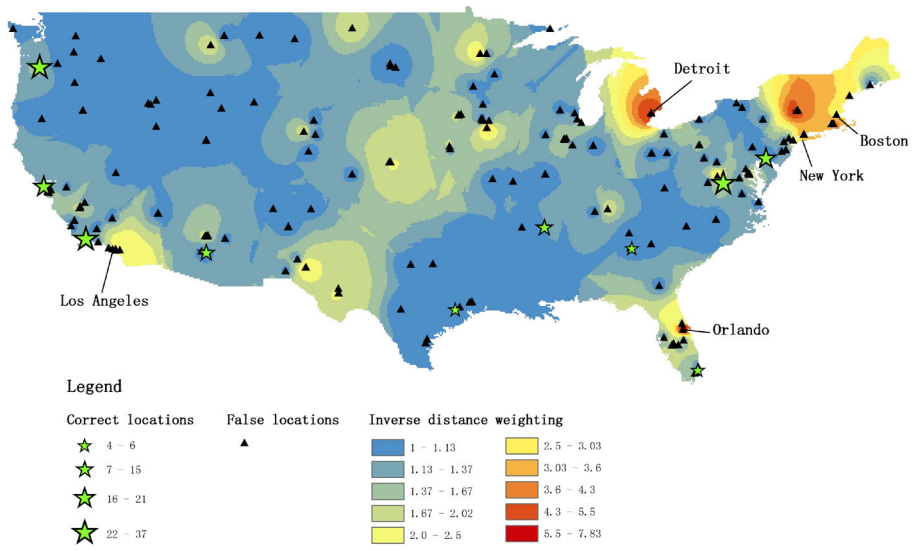

Figure 1 Geographical distributions of correct and false clusters (including anonymous submissions). We use inverse distance weighting (IDW) method [18] to interpolate the distributions. The *Z*-values of the interpolations is the total number of teams submitting the false cluster. The color of the map represents spatial density of the *Z*-values. The locations of correct clusters are marked as green stars with size varying according to the total number of teams successfully locating them. There is no location submitted in Hawaii and Alaska, so above map do not cover those two areas.

Even after removing anonymous submissions, we still observe 10 correct clusters (81 correct locations) and 166 false clusters (213 false locations). Surprisingly, there are 67 clusters (including anonymous submission) located outside the United States, which indicates the lack of even basic verification capability for some teams. The winning submission is the one containing all ten correct locations, submitted by the MIT team at 532 minutes after the challenge was launched. Figure 1 demonstrates locations of correct and false clusters within the continental US, along with the number of teams that submitted them (presented in the interpolated form with warmer color corresponding to the higher number of submitting teams). Given that the areas surrounding official balloons are mainly in light color (blue) in the map, we can assert that correct locations were accurately reported. Moreover, there existed highly confusing false clusters (surrounded with warmer colors) during the challenge.

According to the first panel of Figure 2, false locations and correct locations developed synchronously in the early stage (from 0 to 300 minutes) of the competition, but diverged eventually. We denote a set $X = (1, \ldots, x)$, $x = 43$ of participating teams including the Anonymous team. Moreover, the number of correct locations found is a vector $C = (c_1, \ldots, c_x)$, and false locations as $F = (f_1, \ldots, f_x)$. The ability of verification is denoted as $V = (v_1, \ldots, v_x)$, and $v = c/(c + f)$. Verification ability varies across teams (the correlation between C and F is $r = 0.36$, $p = 0.018$, and this value drops to only 0.09, $p = 0.56$ when excluding anonymous submissions). Moreover, we denote teams' rankings as a vector $H = (h_1, \ldots, h_x)$, and it does not correlate with the verification ability either (the correlation between H and V is $r = -0.1$, $p = 0.53$, and $r = -0.16$, $p = 0.33$ when excluding anonymous submissions), which implies that the verification ability alone does not determine the performance.

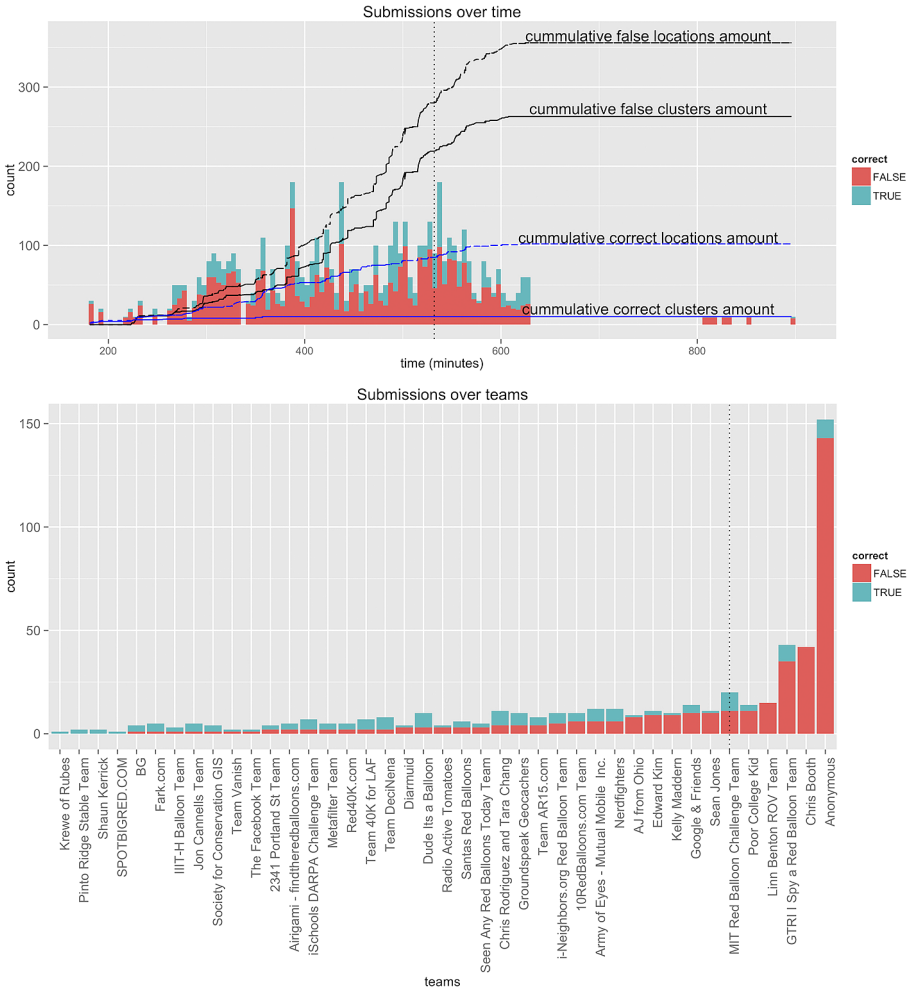

Figure 2 Distribution of submitted locations. The first panel shows the locations submitted over time, and the bin width is 5 minutes. The dotted vertical line at 532 minutes denotes the winning submission. The black solid line indicates the growth of false clusters over time, while the black dashed line indicates the growth of false locations. The blue solid line indicates the growth of official clusters over time, while the blue dashed line indicates the growth of correct locations. All correct clusters were found at 362 minutes after the challenge was launched. The second panel shows the submissions for all 42 registered teams plus one Anonymous team, sorted by the number of false clusters.

During the challenge, multiple teams made submissions that are part of the same cluster, which is called a *suspicious* cluster. This implies that some teams might use similar sources of information. Figure 3 illustrates the network between teams and false clusters. There are a total of 53 suspicious clusters. These false clusters might come from malicious attacks to prevent competing teams from discriminating correct locations from false ones. The lasting time (indicated by color of vertex in Figure 3) of a cluster, which is the time difference between its first and last submission during the challenge, gives an indication of the duration in which the suspicious cluster affected the DNC. As in Figure 3, those long-standing clusters also developed higher betweenness centrality, which means they were so deceptive that most teams were affected by them. In Section A of Additional file 2, a network graph showing the relationships between teams is also listed. In that graph, two teams are linked together if both submitted at least one identical cluster. The high-ranking

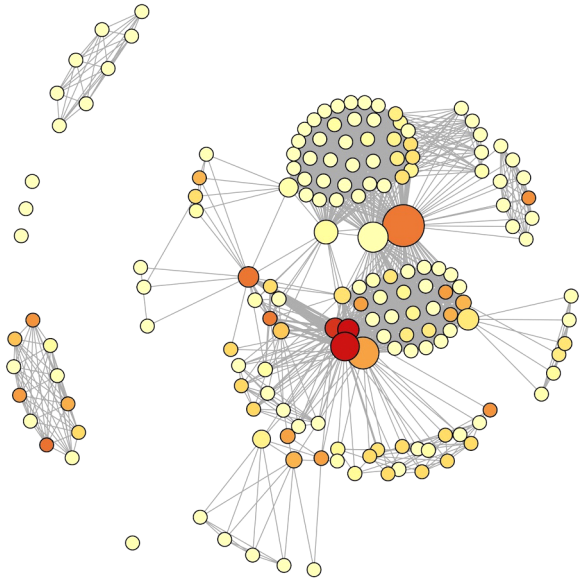

Figure 3 Network graph of false clusters. The nodes represent false clusters, and two nodes are connected if the same team submitted them. The color of each node indicates the lasting time, which is the time difference between its first and last submission. The size of the node indicates the betweenness centrality of that cluster within the graph.

Table 1 Variables of clusters

Variable	Description
$a_{submission}$	Submission count: how many submissions the cluster had
$a_{balloon}$	Balloon count: how many reported locations the cluster had
a_{team}	Team count: how many teams submitted locations belonging to the cluster
a_{appear}	Appearance time: the earliest appearance time of the cluster
$a_{disappear}$	Disappear time: the latest appearance time of the cluster
$a_{lasting}$	Lasting time: $a_{disappear}$ - a_{appear}
$a_{half\text{-}submission}$	Half-life submissions: time elapsed to reach half of all submissions for the cluster. It's the different to the appearance time a_{appear}: if the cluster only has 1 submission, the value is 0
$a_{half\text{-}balloon}$	Half-life balloons: time elapsed to reach half balloons
$a_{half\text{-}team}$	Half-life teams: time it took for the cluster to be detected by half of the teams that report it over the course of the challenge

teams developed higher betweenness centrality, which means they eventually explored the sources used by most of the other teams, and that makes them more vulnerable to attacks.

To analyze how much those potentially malicious false clusters affect the competition and how teams react to malicious attacks, we characterize each cluster (including the correct ones) using a group of variables $a = (a_{submission}, a_{balloon}, a_{team}, a_{appear}, a_{disappear}, a_{lasting}, a_{half\text{-}submission}, a_{half\text{-}balloon}, a_{half\text{-}team})$. A vector $A = (a_1, \ldots, a_i)$, $i = 273$ denotes the features of 273 clusters found during the whole competition. These variables are listed in Table 1. The IDW analyses using these variables are listed in the Section B of Additional file 2.

Figure 4 denotes the distributions of correct and false clusters from all registered teams over 9 variables respectively. The false clusters have smaller $a_{submission}$, $a_{balloon}$, a_{team}, $a_{half\text{-}submission}$, $a_{half\text{-}balloon}$, and $a_{half\text{-}team}$. Moreover, correct clusters appeared earlier and disappeared later than false clusters, with longer lasting time than false clusters (observed from the last row of graphs). This phenomenon may be explained by a number of observations made by DARPA [13]:

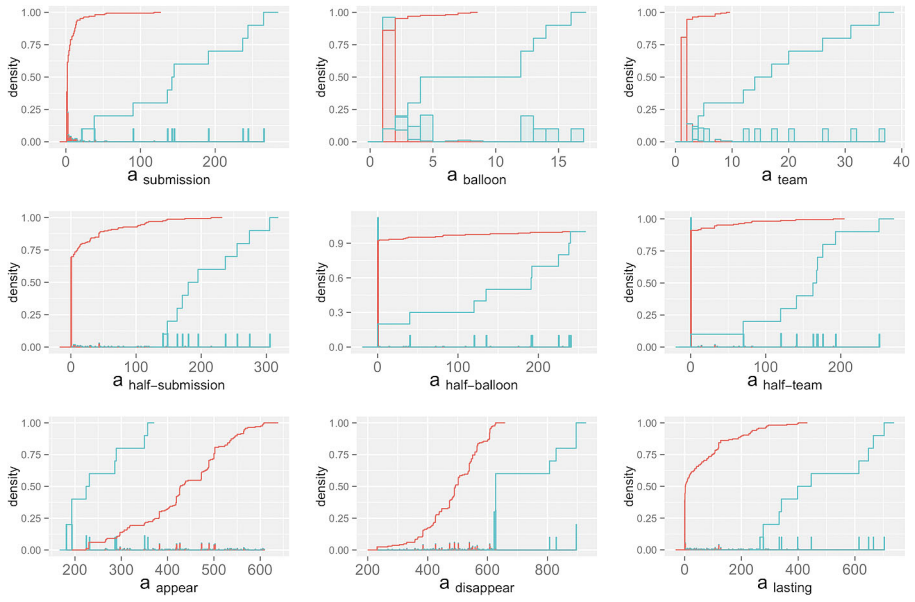

Figure 4 Distributions of correct and false clusters from all registered teams (excluding anonymous) over 9 variables (see Table 1). The red line indicates cumulative distribution function (CDF) of false clusters, while green line indicates CDF of correct clusters. The histograms at the bottom of each panel show distributions of correct and false clusters.

1. Most teams managed to discriminate correct information and false information.
2. Teams changed their strategies over time to acquire more information from more sources, which increases the chances of receiving false information.
3. Malicious attacks were not organized until the white-hot stage, and most of them did not last a long time.
4. There are a few confusing false clusters that have similar characteristics as correct clusters.

Due to a deficiency in the submission data (a total of 545 submissions according to [13]), analyzing the submission behavior of each team was not practical, because some teams only submitted once during the whole competition. Consequently, we use sliding windows (each contains 10 teams) on 42 registered teams, which are sorted by their rankings. We used multivariate logistic regression analysis to find the best model for classifying correct and false clusters for each window. We found that the combination of a_{team}, $a_{half\text{-}balloon}$, and $a_{half\text{-}submission}$ is the best fit model overall (best coefficient of determination R^2, up to 0.83). This R^2 score of each window can be regarded as the verification ability of the corresponding 10 teams. According to Figure 5, better teams did not necessarily have better verification abilities (first panel, the correlation between rankings of window and R^2 is $r = -0.545$, $p = 0.001$). In terms of the locations amount, better teams generally submitted more than worse teams (second panel, $r = -0.565$, $p < 0.001$), except some low ranking teams that submitted a lot. However, the multiplication of locations amount and the verification ability (R^2) significantly correlate with the rankings (third panel, $r = -0.847$, $p < 0.001$).

We use Principal Component Analysis (PCA) [19] to extract the main factors for those 9 variables. The first two factors account for 94.7% of the variance (68.3% for the first factor and 26.4% for the second factor) (Figure 6). The first factor can be regarded as the

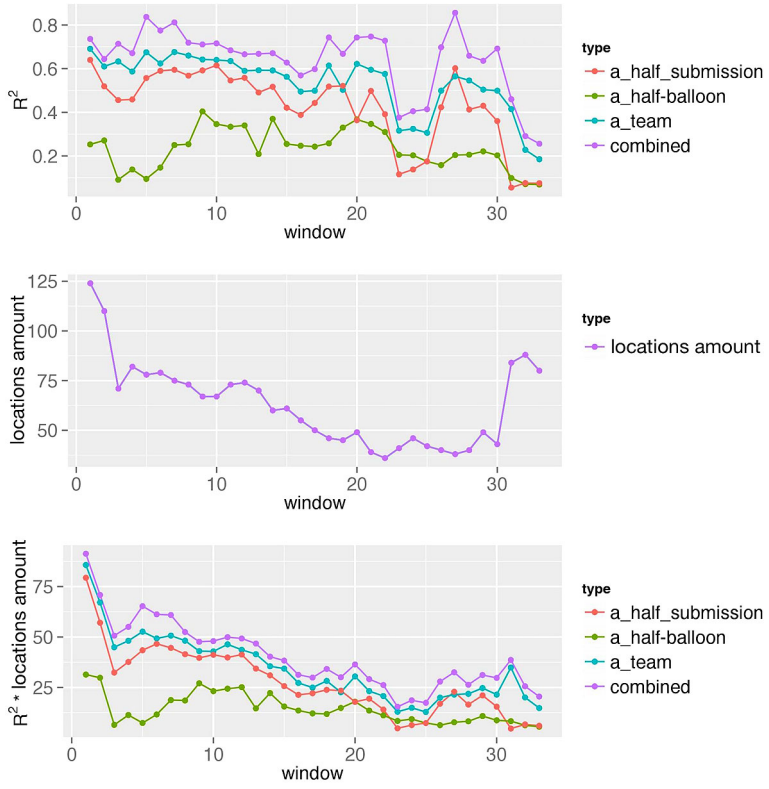

Figure 5 Multivariate logistic regression analysis (a_{team}, $a_{half\text{-}balloon}$, and $a_{half\text{-}submission}$; multiply by locations amount; windows sorted by rankings). The lower the window number, the better the rankings of the teams of the window. The first panel shows the distribution of the R^2 scores over 33 sliding windows. The second panel shows the distribution of the locations amount over 33 sliding windows. The third panel shows the distribution of the multiplication of R^2 scores and corresponding locations amount over 33 sliding windows.

overall verification ability [20] (first panel of Figure 6, the correlation between rankings of window and R^2 is $r = -0.564$, $p < 0.001$). Similar to Figure 5, multivariate regression analysis on three main factors demonstrates that multiplication of locations amount and verification ability (R^2 score) significantly correlate with the rankings (bottom panel of Figure 6, $r = -0.88$, $p < 0.001$).

3 Simulation

3.1 Multi-armed bandit (MAB) problem

According to DARPA [13], teams expanded their sources of information as the challenge progressed, e.g. purchased information from other teams or obtained from Twitter's posts. Retrieving information from new sources can be seen as a form of exploration, while verifying the existing information sources is exploitation. We assume that exploration and exploitation are two competing processes [12], due to limited resource (mainly time) each team has. This implies that at in each trial, teams need to make a choice between submitting the information from what they consider the most reliable sources (exploitation) and submitting from another source (exploration), so this kind of social search problem could be modelled as a MAB problem.

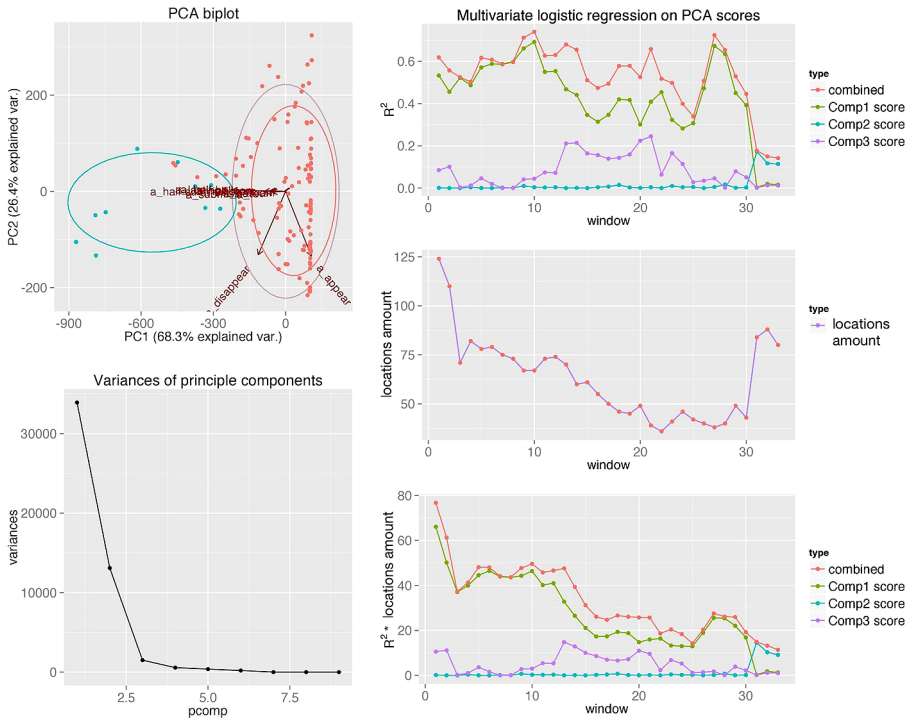

Figure 6 Principal components analysis. The panel on the top left corner shows the biplot of the analysis. The panel on the lower left corner shows the distributions of variances of principle components. The right panel shows the multivariate regression analysis using the first three main components of PCA.

The conventional MAB problem is a problem in which one gambler facing multiple slot machines, each of which has an unknown probability distribution of rewards, needs to decide (a) which machines to play, (b) the order of play, and (c) the number of times to play each machine, to maximize rewards [15, 16]. The player should spend a portion of the limited budget to explore every machine (or some of them) to estimate the distribution of rewards, and then use the remaining budget to exploit the ones with highest expectations.

An adaptive allocation rule to attain the asymptotic lower bound for the regret when the reward distributions are the one-parameter exponential family was proposed by Lai and Robbins in [16]. Based on their work, Agrawal and Hedge [21] expanded the problem and introduced switching cost to the MAB problem attaining the asymptotic lower bound for the regret as well. Switching cost occurs along with the exploration of different machines. It discourages frequent switching, which also applies to a number of practical problems, e.g. oil exploration [22], research and development [23], and website morphing [24, 25]. Other variations of MAB problem have been studied, with various objectives. Hauser et al. [24] explored when the best website morphing time is (switching to another website layout style) to increase consumer's purchase probability. In online shortest path problems, the objectives are to minimize delays occurring in network links which are unknown at first but become more predictable over time [26, 27].

In a social search problem discussed in this paper, the objective is to minimize the time required to find all key information (all correct locations in this study). We propose a general MAB model to solve time-critical social search tasks: Given k people and a set of opinions M, where each person j espouses $s_j \in M$ opinions, a player, who has access to each

of the k opinionated people and has heard opinions W. Consider the following process: a player sequentially selects a person, j, and verifies an opinion in $s_j \in W$. If the information is true, the player receives payoff 1, and otherwise receives nothing. The objective is to minimize W when payoff meets a certain threshold.

In the DNC, this general form of MAB model is extended to have the following properties:

1. The sources are regarded arms. In DNC, trading with other teams is regarded as one kind of sources.
2. The reliability of a source can be estimated by submitting certain pieces of the information provided by the source. In DNC, some teams submitted a single location to validate it through DARPA. It could be an effective submission strategy, if not an optimal one, because submitting more than one could be confusing if the score from the feedback is less than the numbers of locations in the submission.
3. The switching cost $t_{\text{switch}} \times d$ occurs when a team explores d sources and each time it takes t_{switch} to get access to a new source, e.g. negotiation time with other teams when trading locations. We assume that the switching cost is one-off in DNC, which means switching to the explored sources would not generate additional cost.
4. Each submission is a trial. The cost $t_{\text{submission}} \times n$ occurs when a team submits n times and each time it takes $t_{\text{submission}}$ to wait for the feedback.
5. A team receives payoff 1, only if the submission is correct and unobserved before.
6. The objective is to minimize the total search cost $t_{\text{search}} = t_{\text{submission}} \times n + t_{\text{switch}} \times d$ when finding all key information, that is payoff = 10. Moreover, key information could be repeated in different sources.

We consider the following set of strategies, which were previously studied in relation to MAB: ϵ-greedy and its variants, interval estimation (referred as IntEstim in the following), SoftMax, and POKER [28, 29]. As the winning criteria of the DNC is discovering all correct locations, so in an ideal case, where team can submit all 10 correct locations within only 10 trials, the number of trials is 10. However, in reality, the number of submissions is higher than the number of sources, due to the dominating number of false locations over correct ones (Figure 2). In such case, MAB's heuristic algorithms ϵ-greedy and interval estimation strategy are applied as both are proven to be promising strategies [28, 29]. However, we don't consider SoftMax strategy [30] and the POKER strategy [28] and their variants, as the former underperforms other strategies and the latter does not suit in this case where there are more trials than arms [28]. Overall, we test 4 strategies: basic ϵ-greedy, ϵ-first, ϵ-decreasing, and IntEstim.

The ϵ-greedy strategy and its variants have common greedy behaviors where the best arm (the one of highest rewards expectation based on acquired knowledge) is always pulled except when a (uniformly) random action is taken [28]. The basic ϵ-greedy strategy defines a fixed value of ϵ, which is the probability that a random arm is selected in the next trial. The ϵ-first strategy tends to explore in the first ϵ N trials, and exploit the best arms in the remaining $(1 - \epsilon)$ N trials. As the estimation for the rewards distribution of each arm becomes more accurate over time, a fixed ϵ would possibly make the exploration at later stage inefficient. As an improvement, a more adaptive greedy strategy called ϵ-decreasing strategy was proposed, where the value of ϵ decreases as the experiment progresses, resulting in highly explorative behavior at the beginning, but highly exploitative behavior at the end [29]. Different to fixing ϵ in the former two cases, ϵ-decreasing strategy requires

a user to fine-tune the parameter c, which controls the decreasing rate of ϵ, to achieve approximate optimal solution. According to the Theorem 3 of [29], let Δ be the difference between the expectation μ^* of the best arm and the expectation μ of the second best arm. The decreasing ϵ is defined as $\varepsilon \stackrel{\text{def}}{=} \min\{1, \frac{ck}{n\Delta^2}\}$, where k is the number of arms, and n is the number of trials. The larger the value of c, the slower the ϵ decreases, the more exploration is performed.

In an IntEstim strategy, each arm is assigned an "*optimistic reward estimate*" within a certain confidence interval, e.g., 95%, and the arm with highest estimate is pulled [28]. The upper bound of the reward estimation of an arm on step n is computed based on Algorithm 10 in [31]. The confidence level is denoted as z, and the upper bound is defined as follow:

$$ub(\hat{\mu}, \nu) = \left(\frac{\hat{\mu}}{\nu} + \frac{z^2/2}{2\nu} + \frac{z/2}{\sqrt{\nu}}\sqrt{\left(\frac{\hat{\mu}}{\nu}\right)\left(1 - \frac{\hat{\mu}}{\nu}\right) + \frac{z^2/2}{4\nu}}\right) \Big/ 1 + \frac{z^2/2}{\nu},$$

where $\hat{\mu}$ and ν are the observed rewards and the number of times that arm has been pulled by step n. The unobserved or infrequently observed arms tend to have overestimated reward mean, which will lead to further exploration of those arms. The more an arm is pulled the closer the estimate to the true reward mean [28]. There are two reasons causing the upper bound to be large: (1) the arm is seldom pulled, and (2) the observed rewards distribution is good. Moreover, higher confidence level z leads to more exploration [28]. This experiment uses confidence levels ranging from 50% to 99.98% to test the IntEstim strategy (the corresponding z scores of different confidence levels can be found in [32]).

3.2 Experiment settings

In theory, the number of sources during a time-critical social search could be unlimited, since participating teams are free to explore Twitter feeds, Facebook groups, online forums, personal contacts and any other type of sources without restrictions. New sources could be acquired at any stage of the challenge. Teams have no a priori knowledge of the number of sources available. However, the course of the DNC demonstrated that teams accumulate all key information from limited number of sources, which they also trade with each other. To reflect this, and to simplify the experiment, we assume that all information M (correct and false) is provided by a fixed number $k = (20, 40, \ldots, 80)$ of sources that are equally accessible by any team.

Teams have no initial knowledge about the reliability of sources and this knowledge will be gained through submitting the information from them. A set $M = (1, \ldots, m)$, $m = 458$ of unique locations was submitted during the whole competition, so each source contains up to $s_j = \frac{m}{k}$ pieces of information. To simplify the experiment, we assume all sources have equal amount of information. Ten correct clusters sorted by a_{appear} contain 20, 2, 17, 17, 4, 21, 3, 4, 13, 1 locations respectively. Therefore, the set of locations $M = (L_c, L_f)$, where $L_c = (l_1^1, \ldots, l_1^{20}), (l_2^1, l_2^2), \ldots, (^1_{l10})$ is the collection of correct locations, and $L_f = (l_{103}, \ldots, l_{458})$ is the set of the false ones. We assume that one-off switching cost t_{switch} occurs when a team explores a new source. Due to the lack of information about switching cost in DNC, to simplify the experiment, we assume that switching time is the same for each new source and during each simulation run t_{switch} is randomly set to be equal to 5, 10, 15, 20, 25, or 30 minutes. The modified MAB problem is tested in three configurations:

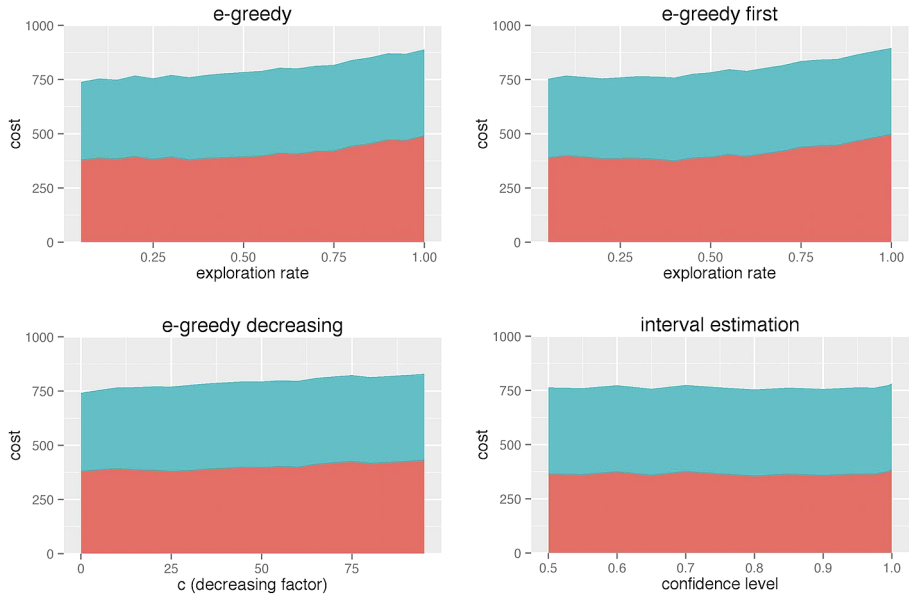

Figure 7 Comparisons between different MAB problem strategies. The blue area indicates the switching cost on average, and the red area indicates the submission cost on average. The overall search time is the sum of those two areas.

I. Locations in M (regardless whether correct or false) are uniformly distributed between k sources;
II. Locations in M (regardless whether correct or false) are normally distributed between k sources, and
III. Correct clusters are set to contain the same number of locations (10 locations of each), so that $L_c = (l_1^1, \ldots, l_1^{10}), \ldots, (l_{10}^1, \ldots, l_{10}^{10})$, and correct locations are normally distributed in k sources.

The setting III is to test the performance of all strategies when all correct information has equivalent appearances.

Since the actual competition lasted 900 minutes, we set the average interval between two submissions as $t_{\text{submission}} = \frac{900}{m} \approx 2$ minutes. A team completes the challenge when all 10 correct locations are successfully submitted, therefore the total search time

$$t_{\text{search}} = t_{\text{submission}} \times n + t_{\text{switch}} \times d,$$

where n is the number of trials, d is number of explored sources, and t_{search} is the score of the team. Each strategy is run 1,000 times to report the average value of t_{search}.

3.3 Results

In a randomized dataset (setting I), all sources tend to have similar reliability. Therefore, exploration oriented strategy or exploitation oriented strategy would not be significantly different. The experimental results confirm this assumption, with no strategy standing out from the others, and the t_{search} converges at approximately 950 minutes ($k = 20$, $t_{\text{switch}} = 20$).

In a normally distributed dataset (setting II), all strategies can achieve the best $t_{\text{search}} \approx 750$ minutes ($k = 20$, $t_{\text{switch}} = 20$), when parameters are properly set (Figure 7). It should

be noted that t_{search} could be as low as 400 minutes when a team explores the most reliable sources during the exploration phase. The ϵ-greedy strategy and its variants could underperform compared to IntEstim if the value of ϵ is not properly set. Similar to the findings of [28], making ϵ decreasing does not improve the performance. The results of the ϵ-greedy strategy and its variants imply that the highly exploitative behavior could possibly lead to lower switching cost and overall better performance, which means teams should focus on the most reliable sources ever found if they adopt ϵ-greedy strategies. However, IntEstim performs well no matter how user defines the confidence level, even though the switching cost is relatively higher than the best settings of ϵ-greedy strategies. Therefore, IntEstim strategy should be adopted in this kind of competition, where some key information appears rarely across sources. We also made submission interval to follow Weibull distribution ($\lambda = 1$, $\kappa \in (1, 5)$, $E(X) = 2$), and the result hold as well. As some correct clusters only contain relatively small number of locations, a team must switch between many sources to collect them if missing in the early stage during the exploration. Therefore, in setting II, the difference of switching cost between the best strategy and the worst one is marginal.

However, in setting III of the simulation, where all correct clusters have equal number of locations, the switching cost dominates the variances of the total search time (see Section C of Additional file 2). A team would probably collect all ten correct locations from a small number of sources. Therefore, the highly exploitative ϵ-greedy strategy and its variants outperform the IntEstim strategy by switching less. Given that the highly exploitative ϵ-greedy strategy and its variants achieve overall promising performance in setting II and III, they should also be adopted in a more general social search problem with unknown rewards distribution.

In conclusion, the results suggest that there would be no universal optimal strategy for time-critical social search tasks of different rewards distributions. Even though the IntEstim strategy outperforms others in the case of DNC, it could generate higher switching cost than the others on average. While in the cases where switching cost is higher than verification cost, the IntEstim strategy could result in an undesired solution. On the other hand, highly exploitative greedy behaviors could guarantee minimum number of switches, while performance is only marginally downgraded. Therefore, in general time-critical social search tasks where rewards distribution and switching cost is usually unknown, we suggest adopting highly exploitative ϵ-greedy strategy and its variants.

4 Discussions and conclusions

To the best of the authors' knowledge, it is impossible to understand the strategies of an individual team given that only a few of them were interviewed after the competition [33]. Moreover, it is not clear how much information each team collected and how reliable this information was. Through the analysis of the submission history, we found that the dominant teams do not necessarily submit the most or have the best verification ability. However, it is the combination of both that leads to success in the competition. When exploration and exploitation are regarded as two competing processes, teams need to balance between exploration of new sources and exploitation of the most reliable ones to gain advantage. As this competition can be seen as a MAB problem, we assume that solutions of other MAB problems could also be effective in this case. Firstly, we propose a general form of the MAB model for handling the time-critical social search problem where multiple information sources are presenting possibly inaccurate information; secondly, we extended

it to adapt to the context of DNC. Agent-based simulations of different strategies are performed to obtain the optimal one for DNC. The result suggests that, in a situation where some key information is rare (known only to a few sources), the IntEstim strategy outperforms the others on average, no matter what confidence level is defined. It also agrees with the findings of other studies [28] that ϵ-greedy strategy and its variants have very similar performance, and making ϵ decreasing would not improve that. On the other hand, if all key information has similar number of appearances, highly exploitative ϵ-greedy strategy and its variants could be the most promising strategies. Given that general time-critical social search problems usually have unknown reward distribution and switching cost, we suggest adopting highly exploitative ϵ-greedy strategy and its variants.

The experiment is performed in only three settings: I - correct locations are randomly distributed across sources, II - correct locations are normally distributed across sources, and III - correct locations have equivalent appearances and they are normally distributed across sources. However, in reality, the distribution of misinformation is unknown beforehand. Therefore, highly exploitative ϵ-greedy strategy and its variants might not work in other cases. It implies that more works need to be done in analysis of the distribution of misinformation in other time-critical crowdsourcing tasks.

Switching cost is unavoidable in practical problems. However, in time-critical social search tasks, little has been done in research about the cost of exploring new sources, or the relationships between switching cost and verification cost. Further analysis about them should be performed.

Even though the result of this study cannot be directly applied in other time-critical crowdsourcing tasks, it provides insights into how to strike a balance between exploration and exploitation. For example, during the crowdsourced manhunt event for the Boston Marathon bombers, the authorities should have observed that misinformation dominated useful information from the very beginning [9], and the self-correcting crowd hardly deterred it. Therefore, attentions should be paid to the discussion threads or Twitter feeds that continually deliver useful information that advanced the search.

Additional material

Additional file 1: Video of submissions over time. The official balloons are shown in green icons, and the false locations are shown in red icons. (mov)
Additional file 2: Supporting materials for Bandit Strategies in Social Search: the case of the DARPA Red Balloon Challenge. (pdf)

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
HC and MC collected and pre-processed the data. HC and MC designed the research. HC, IR and MC analyzed the result. HC was the lead writer of the paper. HC, IR and MC wrote the paper.

Author details
[1]Data61 Unit, Commonwealth Scientific and Industrial Research Organization, Melbourne, Victoria, Australia. [2]Faculty of Information Technology, Monash University, Caulfield, Victoria, Australia. [3]The Media Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Received: 16 December 2015 Accepted: 26 May 2016 Published online: 06 June 2016

References
1. Rutherford A et al (2013) Targeted social mobilization in a global manhunt. PLoS ONE 8(9):e74628

2. Potts L, Harrison A (2013) Interfaces as rhetorical constructions: reddit and 4chan during the Boston marathon bombings. In: Proceedings of the 31st ACM international conference on design of communication
3. Oh O, Agrawal M, Rao HR (2013) Community intelligence and social media services: a rumor theoretic analysis of tweets during social crises. MIS Q 37(2):407
4. Aramaki E, Maskawa S, Morita M (2011) Twitter catches the flu: detecting influenza epidemics using Twitter. In: Proceedings of the conference on empirical methods in natural language processing
5. Munro R (2013) Crowdsourcing and the crisis-affected community. Inf Retr 16(2):210
6. Kryvasheyeu Y, Chen H, Obradovich N (2016) Rapid assessment of disaster damage using social media activity. Sci Adv 2(3):e1500779
7. Bawden D, Robinson L (2009) The dark side of information: overload, anxiety and other paradoxes and pathologies. J Inf Sci 35(2):180
8. Gupta A et al (2013) Faking sandy: characterizing and identifying fake images on Twitter during Hurricane Sandy. In: Proceedings of the 22nd international conference on World Wide Web companion
9. Starbird K et al (2014) Rumors, false flags, and digital vigilantes: misinformation on Twitter after the 2013 Boston Marathon bombing. iConference
10. Boididou C et al (2014) Challenges of computational verification in social multimedia. In: Proceedings of the companion publication of the 23rd international conference on World Wide Web companion
11. Canini KR, Suh B, Pirolli PL (2011) Finding credible information sources in social networks based on content and social structure. In: 2011 IEEE third international conference on privacy, security, risk and trust (PASSAT) and 2011 IEEE third international conference on social computing (SocialCom)
12. Gupta AK, Smith KG, Shalley CE (2006) The interplay between exploration and exploitation. Acad Manag J 49(4):693
13. DARPA. DARPA Network Challenge Project Report. http://www.eecs.harvard.edu/cs286r/courses/fall10/papers/ProjectReport.pdf. Accessed 2016-03-19
14. Smith JR (2010) The red balloon. IEEE Multimed 17(2):2
15. Gittins JC (1979) Bandit processes and dynamic allocation indices. J R Stat Soc, Ser B, Methodol 41(2):148
16. Lai T, Robbins H (1985) Asymptotically efficient adaptive allocation rules. Adv Appl Math 6(1):4
17. DARPA. DARPA Network Challenge. http://archive.darpa.mil/networkchallenge/. Accessed 2016-03-19
18. Shepard D (1968) A two-dimensional interpolation function for irregularly-spaced data. In: Proceedings of the 1968 23rd ACM national conference, ACM '68
19. Jolliffe I (2005) Principal Component Analysis. Wiley Online Library
20. Woolley A et al (2010) Evidence for a collective intelligence factor in the performance of human groups. Science 330(6004):686
21. Agrawal R, Hedge MV (1988) Asymptotically efficient adaptive allocation rules for the multiarmed bandit problem with switching cost. IEEE Trans Autom Control 33(10):899
22. Benkherouf L, Bather JA (1988) Oil exploration: sequential decisions in the face of uncertainty. J Appl Probab 25(3):529
23. Weitzman ML (1979) Optimal search for the best alternative. Econometrica 47(3):641
24. Hauser JR, Liberali G, Urban GL (2014) Website morphing 2.0: switching costs, partial exposure, random exit, and when to morph. Manag Sci 60(6):1594
25. Hauser JR et al (2009) Website morphing. Mark Sci 28(2):202
26. Awerbuch B, Kleinberg R (2008) Online linear optimization and adaptive routing. J Comput Syst Sci 74(1):97
27. Flaxman AD, Kalai AT, McMahan HB (2005) Online convex optimization in the bandit setting: gradient descent without a gradient. In: Proceedings of the sixteenth annual ACM-SIAM symposium on discrete algorithms, SODA '05
28. Vermorel J, Mohri M (2005) Multi-armed bandit algorithms and empirical evaluation. In: Machine learning: ECML 2005. Springer, Berlin, p 437
29. Auer P, Cesa-Bianchi N, Fischer P (2002) Finite-time analysis of the multiarmed bandit problem. Mach Learn 47(2–3):235
30. Luce RD (2012) Individual choice behavior: a theoretical analysis. Courier Dover Publications, New York
31. Kaelbling LP (1993) Learning in embedded systems. MIT press, Cambridge
32. Richland Community College. Stats: Introduction to Estimation. https://people.richland.edu/james/lecture/m170/ch08-int.html. Accessed 2016-03-22
33. Tang JC et al (2011) Reflecting on the DARPA red balloon challenge. Commun ACM 54(4):78