iTag: Incentive-Based Tagging

Siyu Lei, Xuan S. Yang, Luyi Mo, Silviu Maniu, Reynold Cheng

Department of Computer Science, University of Hong Kong
Pokfulam Road, Hong Kong
{sylei,xyang2,lymo,smaniu,ckcheng}@cs.hku.hk

Abstract—In social tagging systems, such as *Delicious*[1] and *Flickr*[2], users are allowed to annotate resources (e.g., Web URLs and images) with textual descriptions called tags. Tags have proven to be invaluable building blocks in algorithms for searching, mining and recommending resources. In practice, however, not all resources receive the same attention from users, and as a result, most tags are added to the few highly-popular resources, while most of the resources receive few tags. Crucially, this incomplete tagging on resources can severely affect the effectiveness of all tagging applications.

We present iTag, an incentive-based tagging system, which aims at improving tagging quality of resources, by incentivizing taggers under budget constraints. Our system is built upon traditional crowdsourcing systems such as Amazon Mechanical Turk (MTurk). In our demonstration, we will show how our system allows users to use simple but powerful strategies to significantly improve the tagging quality of resources.

I. INTRODUCTION

In recent years, collaborative tagging applications such as Delicious and Flickr, have surged in popularity. They allow users complete freedom in describing their shared resources (e.g., Web URLs, images, video or sound clips) by *tagging*: assigning short textual descriptions or *tags*, to resources. Tags are proved to be invaluable resource in value-added user applications, such as URL classification [1], keyword search [2], and query recommendation [3].

High quality tags are crucial for the effectiveness of tag-based applications. To achieve high service quality, resource providers (such as website administrators or dataset owners) need to ensure that the tags for each resource to be relevant, succinct and cover the various aspects of the resource. However, it is hardly the case in practice. The reasons are twofolds. Firstly, since taggers (the persons who tag resources) are often casual web users, the tags given to a resource by a tagger, are often *noisy* and *incomplete* - they may contain tags that are typos or are irrelevant to the resource (noisy); and they may only describe some of the many aspects of the resource (incomplete). Secondly, in collaborative tagging systems, taggers are free to choose the resources that they wish to tag, thus most tags are directed to a small number of highly popular resources [4].

The above issues lead to a general problem in collaborative tagging applications: popular resources are more likely to have a greater number of tags and hence a greater chance to have high tagging quality, while on the other hand, relatively unpopular resources have a greater chance to have low tagging quality [5].
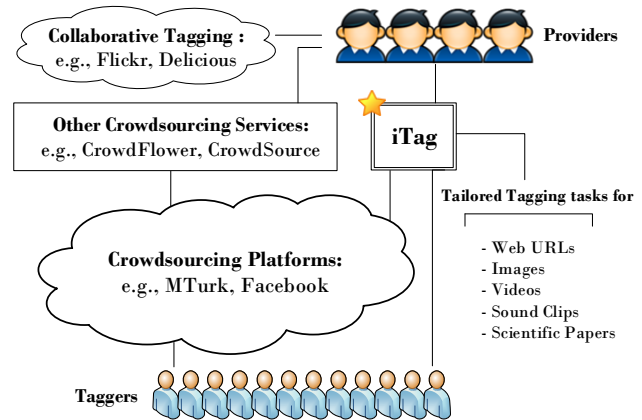
Fig. 1. Relationships between iTag and other platforms and services.

This suggests that, in order to obtain good tagging datasets, resource owners need good strategies for directing taggers to low-quality resources.

Recently, crowdsourcing systems such as Amazon's Mechanical Turk (MTurk) establish effective platforms for resource owners to improve tagging quality. They provide tools for assigning specific tagging tasks to workers, while also incentivizing them. However, such systems have a few limitations:

- Providers need to specify the exact number of tagging tasks to assign, which may be hard for them to decide.
- There is no clear way for providers to prioritize the resources which are sent to workers, and therefore it's impossible for providers to directly control the quality of tagging tasks.

These two issues are especially crucial when providers act under budget constraints.

To solve these problems, we present iTag, an incentive-based tagging system, which is built upon crowdsourcing marketplaces such as MTurk. Essentially, iTag serves as an agent that is devotedly designed for providers with tagging requirements to better allocate their resources, monitor processes and gain the best tagging quality. More specifically, in iTag, providers can upload their resources (e.g., Web URLs, images, videos, sound clips, scientific papers) in need of tagging quality improvement, provide incentive budgets, and choose which platform to use (as scientific papers resources will highly likely be getting better tags with taggers from scientific communities other than MTurk). Using tagging statistics in the dataset and based on the budget, iTag further helps providers to decide the best allocation strategy for the resources, along with monitoring the

projected quality gains. Then, based on the chosen strategy, taggers are assigned resources to tag. We remark that iTag can be extended to other platforms such as social networks to realize incentive-based tagging. For example, [6] builds crowdsourced searching systems on Facebook, the same idea can also be applied to iTag.

Although other crowdsourcing services such as CrowdFlower and CrowdSource have their tagging solutions for providers (they also have solutions to other tasks such as search relevance and sentiment analysis), their only way to control the tagging quality is by limiting tasks only to pre-qualified workforce. They neglect the "noisy" and "incomplete" problems of the usual tagging actions as described in [4], while iTag focuses solely on tagging tasks, and has systematically studied and defined the quality of tags that has never been done before. Beside that, iTag provides the following benefits:

- It offers providers the access to choose the best crowdsourcing platform that is most suitable for their needs.
- It provides several simple but close to optimal strategies for providers, and then automatically and dynamically determines which resources to tag next so as to achieve the best quality.
- Providers can monitor in real-time the quality improvement in their resources, and change allocation strategies if they are not satisfied with the current tagging progress.

The relationships between iTag and other crowdsourcing platforms and services can be founded in Figure 1.

II. MODEL AND STRATEGIES

Data Model. Let $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$ denote a set of n resources, and $\mathcal{T} = \{t_1, t_2, \ldots, t_m\}$ be a set of all possible tags. A *post* is a nonempty set of tags assigned to a resource by a tagger in one tagging operation. The *post sequence* of a resource r_i is the sequence $(p_i(1), p_i(2), \ldots)$, where $p_i(k)$ $(k \geq 1)$ is the k-th post received by resource r_i.

Tagging Quality. In [4], we establish a quality metric $q_i(k_i)$ for a specific resource r_i with k_i posts, which tells how accurate are the tags in describing the resource. The definition of $q_i(k_i)$ is based on the stability of relative frequency distributions (or *rfds*') of the tags given to r_i. For $\mathcal{R}$, the tag quality is defined to be the average tag quality over all the resources: $q(\mathcal{R}, \vec{k}) = \frac{1}{n} \sum_{i=1}^{n} q_i(k_i)$.

Incentive-Based Tagging. Given a set of resources $\mathcal{R}$ where each resource $r_i \in \mathcal{R}$ has got c_i posts, $\vec{c} = (c_1, \ldots, c_n)$, and a budget of B tagging tasks, our goal is to find an assignment $\vec{x}$ $= (x_1, \ldots, x_n)$, $\sum_i x_i = B$, where x_i is the number of tagging tasks assigned to resource r_i, such that $q(\mathcal{R}, \vec{c} + \vec{x}) - q(\mathcal{R}, \vec{c})$, i.e, the quality improvement, is maximized.

Strategies. Our system solves this *incentive-based tagging* problem with a multi-step "choose resources - update model" framework, as shown in Algorithm 1. The framework's functioning principle is as follows.

As long as there is a budget left, a set of resources R_c is selected by CHOOSERESOURCES(). Then, for each $r_i \in R_c$, a tagging task is assigned to a tagger (Step 4). After the task is

---

Algorithm 1 Strategy Framework

Require: Budget B, Resources $\mathcal{R}$, Initial no. of posts $\vec{c}$
1: **for** $i \leftarrow 1$ to n **do** $x[i] \leftarrow 0$
2: **while** $B > 0$ **do**
3:     $R_c \leftarrow$ CHOOSERESOURCES()
4:     assign R_c to taggers
5:     $\forall r_i \in R_c.\ x_i \leftarrow x_i + 1,\ B \leftarrow B - 1$
6:     UPDATE()
    return $\vec{x}$

---

completed (Step 5), UPDATE() is called to update the statistics and the quality scores.

We have the following allocation strategies: Free Choice (FC), Fewest Posts First (FP), Most Unstable First (MU) and Hybrid Strategy (FP-MU), which differ only in the implementation of CHOOSERESOURCES(). Table I shows their short descriptions (Des), along with their advantages (Pro) and disadvantages (Con).

Name	Characteristics
FC	**Des:** Let taggers freely choose resources to tag **Pro:** Get taggers' preferences and popularity of resources **Con:** May not improve tag quality of $\mathcal{R}$ significantly
FP	**Des:** Prioritize resources with fewest posts **Pro:** Reduce the number of resources with low tag quality
MU	**Des:** Prioritize resources with most unstable *rfds*' **Pro:** Increase the number of resources that can satisfy a certain quality requirement
FP-MU	**Des:** use FP first, then use MU **Pro:** Most effective in improving tag quality of $\mathcal{R}$

TABLE I
THE TASK ALLOCATION STRATEGIES.

More details about the model and definitions, along with examples and extensive experiments, can be found in [4].
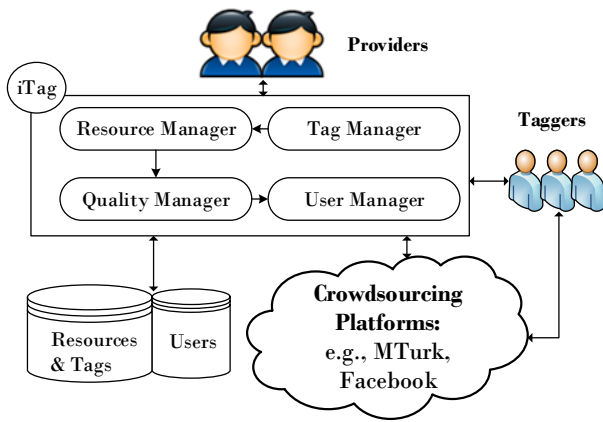
III. SYSTEM OVERVIEW

We begin by presenting the architecture of the iTag system in Figure 2. The system is implemented in PHP and Python, using a MySQL database running on an Ubuntu 12.04 operating system. iTag provides functionality for both providers and taggers, as we will detail next.

A. Provider Functionalities

The first step resource providers must take is to upload resources to the system. Resources can be Web URLs, images, videos, sound clips or scientific papers and their tags can be of low quality or even missing. The resources are then managed by the **Resource Manager**, which is in charge of controlling the operations on resources and their related tags, and is responsible for storing resource and tagging information.

After providers assign a budget for their quality improvement objective, the **Quality Manager** receives the budget together with other resource information, creates a *Project*, and uses the platform that has been chosen by the provider, and executes the best strategy to allocate resources to taggers. It will also constantly provide feedback to the provider during the run of the strategy and allow them to change the strategy if necessary.
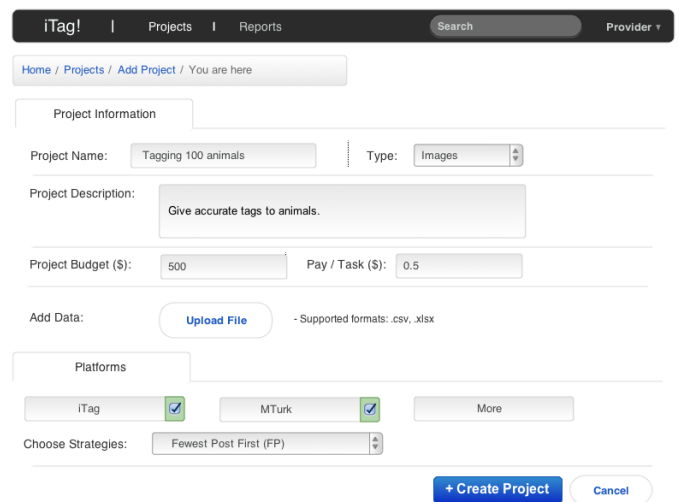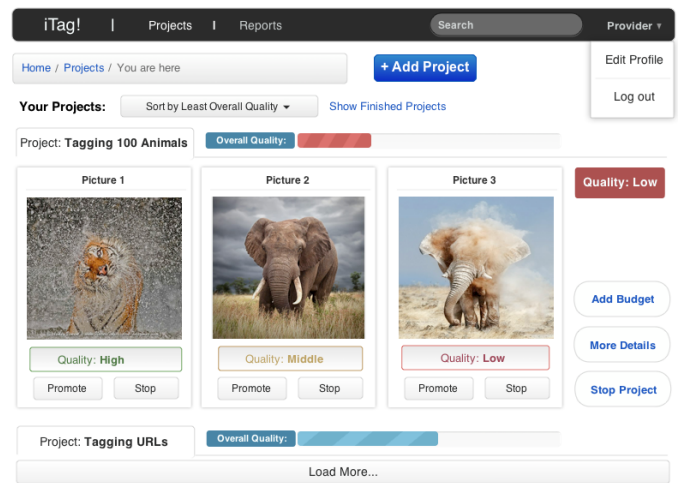
Fig. 2.   iTag architecture.

Fig. 3.   Main provider UI.

Fig. 4.   Add Project.

The provider's and taggers' profile information is handled by the **User Manager**, which also tracks their approval rate, which is the ratio of providers approving the tags of a given tagger, and on the tagger side, the ratio of taggers approving a provider. The role of this approval process is to avoid two undesired outcomes: (i) taggers which provide low-quality tags to resources on a consistent basis, and (ii) providers which hold back on approving tags, thus delaying the payment of incentives. This way, providers can approve good tags, while taggers approve of good and reliable tagging providers. User Manager also guarantees that the approval rate of taggers from crowdsourcing platforms are at a reliable level.

Figure 3 shows the main UI of providers, where projects are listed and can be sorted according to some rules (e.g., tagging quality). The providers can upload resources by clicking *Add Project* button. They can also monitor the quality information of their projects directly. If the quality has been good enough, providers can stop the project, minimize their budget invested and also export resources with the desired tags. Otherwise, providers may add budget to the project, or click *More Details* button to get access to the details of the project. Providers can also manipulate single resource by further decide to invest more on those of low quality by using the *Promote* button (and thus ensuring that the resource will be chosen by the next CHOOSERESOURCES() step), or stop investing certain resources of good tagging quality by pressing *Stop*.

Figure 4 shows specifically how we can add a project. Providers need to specify name, type, description, budget and pay/task of the project, and click *Upload File* to upload resources with possible tags. Also, providers determines the proper crowdsourcing platforms to use. After providers have chosen the platform, iTag will help push the tasks to the platforms using their APIs. We will help providers choose the best strategy given the current resources and tags statistics, while providers can also change the choice if necessary.

Additionally, Figure 5 displays the project details screen, in which the change of quality score is shown, and it helps providers decide whether it is necessary to switch to another strategy. Providers may also choose their favourite crowdsourcing platforms.

Finally, providers can access the detailed quality metrics of an individual resource, as shown in Figure 6. They are able to see tags of resources and their frequencies, and even the quality evolution of the resource. The *Notification* section reminds providers of the latest tagging (allowing them to approve or reject the tagging by clicking the *Approve* or *Disapprove* button) as well as changes in the quality status of resources.

B. Tagger Functionalities

iTag has seamless connections with traditional crowdsourcing platforms. For example, iTag can push tagging tasks according to the selected strategy to MTurk with the help of MTurk APIs. Taggers can therefore work on tasks directly from MTurk, from which iTag will then aggregate results.

The system allows taggers to either choose projects with high pay per task or projects from providers with good approval rate. Once taggers have decided on a project to join, they are assigned resources to tag, as decided by the strategy or the project provider.

The linking of tags to resources is handled by the **Tag Manager**, after the desired resource has been tagged. The **Quality Manager** will then offer the unit of incentive to
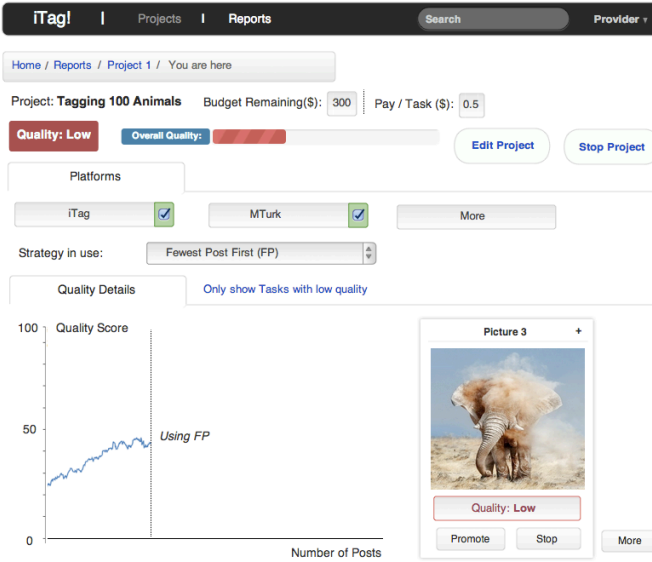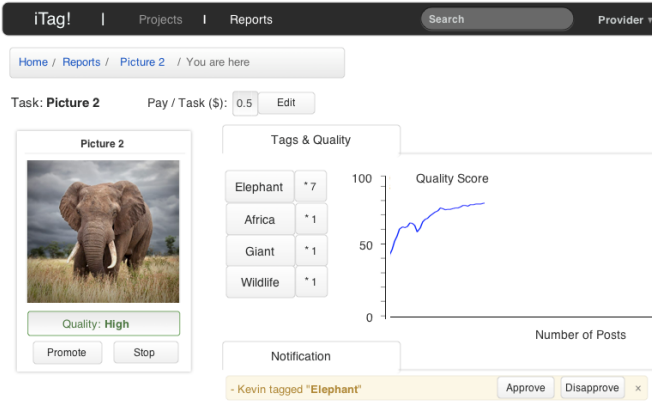
1188

Fig. 5.   Project Details.

Fig. 6.   Single Resource details.

Fig. 7.   Project selection screen.

Fig. 8.   Tagging screen.

taggers, once a tag has been approved by the provider. As in the case of the providers, the **User Manager** will also keep track of the ratio of approved tags as the tagger approval rate.

The main tagger screen is shown in Figure 7. It shows project information such as the name and the approval rate of the provider, and the incentive for tagging one resource. Once a project has been chosen, the access to the resource tagging screen is provided by the *View in Detail* button. Figure 8 illustrates the tagging screen. Taggers can inspect the information of the resource and add tags to a resource by clicking *Add Tag*. Taggers can also inspect their tags pending approval, and view their historical tagging data.

IV. DEMONSTRATION DETAILS

The demonstration will consist of two parts: one in which we will show how the strategies behave on a real Delicious dataset, and one in which demonstration attendees can participate in real-time, both as providers and taggers.

Real Dataset. We have prepared all tagging data for Web URLs from Delicious in the year 2010. We further consider the data before February 1st 2007 as the tagging data of providers, and use the remaining data to evaluate
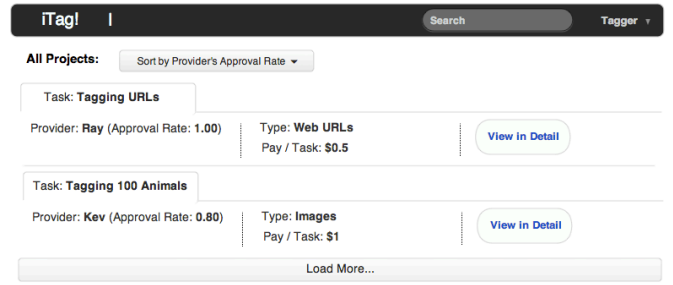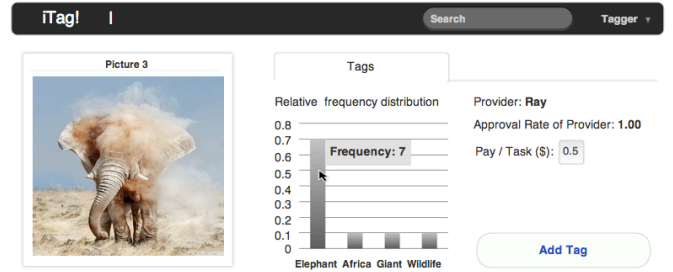
our allocation strategies. We demonstrate in our system how different allocation strategies affect the tagging quality, and compare them with the optimal allocation strategy.

Audience Participation. The audience can participate either as a provider or as a tagger. Taggers can choose their tagging tasks to complete, based on the incentives allocated by the providers' chosen strategy. Their tagging behavior can be validated by the provider if the tagging is deemed of good enough quality, enabling them to earn their reward.

As providers, they can choose from several prepared workloads to publish as resources. They can then establish their desired budget and choose a predefined strategy. In the run of the strategy, they can promote or stop individual tasks, monitor the quality gains and even change the strategy in place if they find it not to behave as desired. The taggers can be either real audience members, or simulated taggers in case there is not enough audience participation on the taggers' side.

REFERENCES

[1] D. Ramage, P. Heymann, C. D. Manning, and H. Garcia-Molina, "Clustering the tagged web," in *WSDM,*, 2009.

[2] B. Bi, S. Lee, B. Kao, and R. Cheng, "CubeLSI: An Effective and Efficient Method For Searching Results in Social Tagging Systems," in *ICDE*, 2011.

[3] J. Guo, X. Cheng, G. Xu, and H.-W. Shen, "Structured Approach to Query Recommendation With Social Annotation Data," in *CIKM*, 2010.

[4] X. Yang, R. Cheng, L. Mo, B. Kao, and D. Cheung, "On Incentive-Based Tagging," *ICDE*, 2013.

[5] S. Golder and B. Huberman, "Usage Patterns of Collaborative Tagging Systems," *Journal of Information science*, 2006.

[6] A. Bozzon, M. Brambilla, and S. Ceri, "Answering search queries with crowdsearcher," in *Proceedings of the 21st international conference on World Wide Web*.   ACM, 2012, pp. 1009–1018.