Global Budgets for Local Recommendations

Thomas Sandholm, Hang Ung, Christina Aperjis, Bernardo A. Huberman
Social Computing Lab, HP Labs
Palo Alto, CA 94304, USA
{thomas.e.sandholm, hang.ung, christina.aperjis, bernardo.huberman}@hp.com

ABSTRACT

We present the design, implementation and evaluation of a new geotagging service, *Gloe*, that makes it easy to find, rate and recommend arbitrary on-line content in a mobile setting. The service automates the content search process by taking advantage of geographic and social context, while using crowdsourced expertise to present a personalized feed of targeted information ranked by a novel geo-aware rating and incentive mechanism.

Users rate the relevance of recommendations for particular locations using a limited, global voting budget. This budget is, in turn, increased by accurately predicting local content popularity. One of the key goals of our mechanism is to encourage ratings, and in an evaluation of the live system we found that the rating to click ratio was 107 times higher than the ratio for videos on YouTube, 34 times higher than the ratio for applications on the Android Market, and 3 times higher than the ratio for Web pages on Digg.

To investigate whether our mechanism also had qualitative effects on the ratings we conducted a number of experiments on Amazon Mechanical Turk, with 500 users, comparing our mechanism to the de-facto 5-star ratings commonly in use on the Web. Our results show that budgets improved the ranking and incentives improved the aggregate rating of a series of location-dependent Web pages.

Categories and Subject Descriptors

H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces

General Terms

Design, Economics, Experimentation

1. INTRODUCTION

Thanks to advances in mobile browser technology and connectivity, the Web platform is becoming ubiquitous across a wide range of devices beyond desktops and laptops including mobile phones, smart phones, slates, netbooks and even printers.

RecSys2010, September 26–30, 2010, Barcelona, Spain.

However, devices with smaller visual real estate, more limited bandwidth, and restricted input mechanisms are ill suited to sift through large amounts of information and content typically presented on the Web, e.g., in search results and map overlays.

Now, consider that more user-contributed content and more sophisticated information extraction techniques have led to an information overflow even for traditional Web platforms. It is then clear that more efficient filtering and aggregation techniques are needed for the mobile Web. One efficient technique to automate the filtering process is to sense the context of the information search from the device or application. Another is to filter the content based on popularity in different demographics or social networks.

Location Based Services (LBS) [1] address the general problem of providing location-aware context filtering. Content is typically filtered based on distance, assuming a physical address of a Point of Interest (POI) and equal popularity among recommended items. In our work we want to relax both of these assumptions to allow users to recommend and rate arbitrary Web content in arbitrary locations, and in that sense construct something very similar to a traditional search engine on the surface, but one that is powered by the crowd and that is highly geo-sensitive.

The issue of many recommendation systems based on crowd ratings today is that they are either too sparse to provide any meaningful aggregate guidance, or too opaque to validate or filter the results based on trusted users. These two features are crucial when filtering today's flood of location-dependant information. For example, Yelp [2] may rate a restaurant very poorly based on just one guest who was unhappy with the experience, or conversely Google may rank the restaurant highly because of a number of hidden metrics and recommendations. So which recommendation should a particular user then choose?

There are two possible ways to tackle this *user-controlled* versus *objective* trade-off problem: either a search engine is made more transparent by adding more meta data, or a rating system is extended to improve the quality and quantity of ratings. Google and other traditional search engines have chosen the former approach [12], whereas in this paper we focus on the second approach, which has been less explored in the literature and in real systems.

We extend the existing state-of-the-art LBS work by introducing novel rating, ranking, and incentive mechanisms that take both social and geographic context into account to estimate the popularity of on-line content, such as Web pages.

Systems such as Yelp and Urbanspoon [3] try to limit the influence

[1] e.g. `http://{mobilizy,layar,gowalla,geodelic,loopt,mypebblebox,foursquare,where}.com`
[2] `http://yelp.com`
[3] `http://urbanspoon.com`

of destructive users by forcing everyone to sign up and leave a text comment with their rating. This extra contribution burden limits the number of ratings, particularly for less known providers. Another common way of tackling the problem is to restrict users to a single vote on a very limited scale (e.g., 1 to 5 stars). This again limits the information that can be inferred, leading to possibly less accurate or simply fewer ratings.

To address this problem we have designed a geographic information economy that aggregates votes restricted by per-user voting budgets. The popularity of Web content is determined by how many aggregate votes a URL pointing to that content has received within a search radius away from a location specified by the user (or sensed by the user's device).

A small business with a limited budget may in this way receive very high prominence in a local area, due to the fact that users of our service could specify an epicenter and a radius of the search that does not include competitors.

Given that our recommendation approach incorporates both social as well as crowd contribution factors that are hard to evaluate solely in a lab setting, we have made our implementation available to the public on a large number of diverse platforms, and we are continuously studying the usage to refine our mechanism. Because of this symbiotic relationship, we present both the mechanism and the system in this paper.

The contributions of this paper include:

- a novel budget-based recommendation mechanism for social and geographic filtering,
- an end-user system, called *Gloe* [4], that implements this recommendation mechanism, and
- a series of evaluations and experiments showing that (i) contribution incentives can improve the quality of individual ratings, (ii) the budget mechanism can extract more accurate aggregate rankings, and finally (iii) the combination of incentives and budget-based ratings in a real system exhibits substantially more ratings per content consumed than similar rating systems.

This paper is organized as follows; in Section 2, and Section 3 we discuss related work, and some motivating examples. In Section 4 we present the underlying model of our ranking and incentive mechanisms, and describe the design behind the implementation of the Gloe service. In Section 5, and Section 6 we evaluate the system and the mechanism using live traces and end-user experiments on Mechanical Turk. Finally, we summarize our findings and conclude in Section 7.

2. RELATED WORK

Related work falls into five broader research areas: *recommender systems*, *crowdsourcing*, *geographic and mobile search*, *paid search* and *incentive-based mechanisms*.

2.1 Recommender systems

Our work relates to the use of contextual information in recommender systems. Adomavicius et al. [1] proposed a general framework to extend existing collaborative filtering algorithms through a multidimensional approach capable of leveraging any contextual information. Others studied the specific use of tags [30] or social networks [13, 10] and showed that both yield better recommendations. Although we also exploit tags and social links, our approach focuses on location. Furthermore, Gloe does not use collaborative filtering techniques but relies on a budget-based rating mechanism.

[4] http://hpgloe.com

2.2 Crowdsourcing

The general concept of *crowdsourcing* and *the wisdom of crowds* was first articulated by Rheingold in [24] and was extensively studied subsequently (e.g., [26, 15]). The idea is that a large group of people may provide more accurate information than a few experts given some aggregation constraints. An example is the Google PageRank mechanism [6], whereby the crowd (Web page providers) indirectly give their estimate of how popular a Web page is by adding links to it from their own pages. One issue with this approach is that only Web page providers' estimates are used, not visitors' estimates. Furthermore, the (by design) implicit rankings may be offset by Web site owners trying to explicitly boost their own ranking [11]. Explicit crowd rating systems address some of these issues, such as the news aggregation site Digg [5] and the social bookmarking service Delicious [6]. However, these systems lack an incentive mechanism to govern the quality of the ratings, and rely heavily on who is most connected in the social graph of the service [28, 29].

2.3 Geographic and mobile search

One may claim that entering a geographic keyword term in a traditional search engine would mimic the behavior of geographically aware search services, such as LBS [25]. There are a number of issues with this approach. First, the geographic search term needs to be explicitly mentioned on the pages, second there is no notion of geographic scope or distance, and third the global ranking (e.g., PageRank which is a global metric) of a page may be very different from the local ranking. A number of *information retrieval* and *data mining* efforts have been proposed to address these issues [2, 7, 3]. These efforts rely on crawling and indexing of Web pages, and do not take social networks nor explicit visitor ratings into account.

Mobile device search customization was addressed in [19, 20], where the conclusion was that recommendations and automated filtering on popular search terms within the current geography were the keys to improving the mobile device search efficiency, given the more limited input mechanisms, e.g., a soft keyboard on a small screen. This work did not leverage the social context or discuss the importance of crowdsourcing to provide these recommendations.

2.4 Paid search

Although Google takes great pride in the *organic* search results being the *true* rankings of Web pages [6] not influenced by commercial interest, the fact is that the main revenue stream comes from their paid search feature *AdWords* [11]. So the success of Google is to be found in the ingenious combination of paid and organic search (e.g., minimizing intrusion and maximizing relevance), as many other search engines have tried to monetize search unsuccessfully [6]. AdWords, however, has the same limitations as Google PageRank in general: it is focused on Web page providers bidding on their content, and visitors may only express their approval by clicking on the links. The general lack of transparency in how AdWords works is not an oversight but rather a design choice (see e.g. the sections on QualityScore in [11]); since the scheme would be defeated if Web page providers could figure out how to bid less to get more prominence for worse content. One reason for this is that the GSP auction mechanism used is known to not be fully truth-telling [8]. This lack of transparency has ripple effects on how the results may be filtered. In particular, users are not aware of how geographic and social network popularity affect the results.

[5] http://digg.com
[6] http://delicious.com

2.5 Incentive-based mechanisms

Bhattacharjee and Goel make a case for sharing the revenue generated by ranking and recommendation systems with users, as encouragement to provide useful feedback and present an incentive based ranking scheme [5]. However, their theoretical analysis is not complemented by an implementation study. Furthermore, our mechanism is different in that novel contributions and geographic coverage are taken into account (see Section 4.1).

Using the crowd to predict future events by aggregating opinions in a market is the general idea behind *information markets* [14]. Information markets are speculative markets created for the purpose of making predictions, and users are incentivized to report their beliefs through monetary incentives. Scoring rules are used to elicit and evaluate the probabilities users assign to future outcomes in reports. Market mechanisms are used to allow trading of reports so that individuals with more accurate information than what is present in the current system may gain from arbitrage [14]. A number of experimental studies show that information markets work well in practice (e.g., [23, 9]).

Our approach borrows many concepts from information markets, but is not a traditional market since no trading takes place. However, if the current economic currency used is mapped to real currency in an exchange, similar trading scenarios would be possible. Our key extension to traditional information markets is the focus on geo-sensitive and social-network sensitive predictions. In this work we are mostly concerned with incentives that increase not only the accuracy of existing items, but also the number of ratings on new items in new geographies (see Section 4.1).

Lastly, our budget mechanism also shares some characteristics with reputation systems [18] where users acquire trust from the community through good behavior and good ratings received from other users. In our case there is no reputation ratings *per se* but since good behavior (predicting accurately) leads to increased budgets, it empowers users with more influence and thus meritocratic status.

3. MOTIVATING EXAMPLES

To demonstrate that many systems relying on explicit ratings have a very low ratio of ratings to clicks (here, content views or downloads), we study YouTube [7], the Android Market [8] and Digg. The first two are studied because they expose both clicks and ratings to the users, and the latter was chosen because the general rating mechanism and the content rated are similar to our approach.

3.1 YouTube video ratings

We used data obtained from the study on YouTube in [27]. The data comprises a sample of $83,702$ videos obtained while tracing recent uploads for 2.5 days in 2007. The videos were studied for three months from the time they were posted. Using the snapshot after 3 months, we measured the *rating ratio* for all videos, defined as

$$\bar{\mu} = \frac{1}{n}\left(\sum_{i=1}^{n} log_{10}(r_i/c_i)\right), \quad (1)$$

where n is the number of items, r_i is the number of ratings on item i, and c_i is the number of clicks on item i (in this case video views). We use log differences to measure relative as opposed to absolute differences given that the values may span multiple orders of magnitude. This also had the effect of transforming the metric into normal distributions for all our data sets, which simplifies statistical testing. Furthermore, to calculate reliable rating ratios we only considered content that has been viewed or rated more than 3 times [9]. We found that $\bar{\mu} = -2.165 \pm .004$ [10] in this sample, indicating that there are more than two orders of magnitude more clicks than ratings on a video on average in YouTube.

3.2 Android Market application ratings

As another reference point we studied the same rating ratio for applications on the Android market in March 2010. In this case we obtained a sample of 9006 applications (of a total of about $34,000$[11]) by querying the market for popular search terms. A click in this case is an application download. Since only ranges of the numbers of downloads are available (and not exact numbers), we chose the lower bound on the range, and ignored the lowest range (applications with less than 50 downloads). We found $\bar{\mu} = -1.666 \pm .009$, which thus can be considered a conservative approximation.

3.3 Digg Web page ratings

The Digg service is designed to engage users in rating Web pages. If enough users rate or *digg* a Web page it will be promoted to the front page, and it is then more heavily exposed to visitors (and thus offers more robust statistics). As our data we took two samples, one from all stories (Web pages) submitted to digg between Jan 2009-Apr 2010 ($2,035$ with more than 3 ratings or clicks from weekly samples) and one from all promoted stories ($6,000$) during the same period. For the submitted sample $\bar{\mu} = -0.548 \pm .031$, and for the promoted sample $\bar{\mu} = -0.619 \pm .017$.

Given that items have on average about $4-146$ ($10^{.548}-10^{2.165}$) times fewer ratings than clicks it is clear that there is a large amount of untapped opinions that could have improved the understanding of the overall quality perception or popularity of an item across all its users. Addressing this untapped knowledge is one of the key motivators for our work.

4. SYSTEM

4.1 Model

In this section we describe the model underlying our system, *Gloe*.

Providing ratings

Each user has a limited budget, B, that can be used to rate Web content (anything that can be retrieved with a URL). The content that is rated may have been previously recommended to the user or the user may rate new content to make it show up in subsequent recommendations. Users who place bids on existing recommendations are hereafter referred to as *voters* and users who place bids on new content are referred to as *contributors*. Budgets are global in the sense that the same budget is used to vote on and contribute content in any location.

A *content rating*, is a 5-tuple $\{c, t, u, w, b\}$, where c is the geographic coordinate expressed as a latitude, longitude pair, t is a tag associated with the rated content, u is the user that is making the rating (by voting or contributing), and w is the title and URL pointing to the Web content. A bid b, which may be both positive and negative, expresses the weight or value of the rating. The bid must satisfy $|b| < B$. After the rating is made, the user's budget is

[7] `http://youtube.com`
[8] `http://market.android.com`
[9] and rated as well as viewed at least once
[10] 95% confidence bound
[11] See e.g., `http://www.androlib.com`

decreased by the magnitude of the bid, i.e., the user's new budget is $B' = B - |b|$.

Each rating is also associated with the time, T, when the recommendation was made. Furthermore, the c coordinate space is clustered in geographic areas, a.

A tag, t, is comprised of a hierarchically organized list of arbitrary strings $\{s_1, s_2, ..., s_n\}$ representing categories and sub categories chosen by the user.

Obtaining recommendations

To obtain a list of recommendations the user specifies a *content query* comprising a 5-tuple $\{c, t, r, h, U\}$ where c is the coordinate from which to search, t is a tag filter, subsequently referred to as a *channel*, $\{s_1, ...s_k\}$ matching the first k tag strings in the ratings previously made, r is the radius within which to search (from the coordinate c), h is the maximum number of search results to return, and $U = \{u_1, u_2, ...u_j\}$ is a set of users to filter the results on. If U is empty, ratings from all users will be returned.

The system matches all ratings within the search radius and returns a list ordered by bids aggregated by URLs. Thus, the most highly ranked result is the one that users spent the largest budget amount in aggregate to recommend (within the search radius). Each result item contains the 4-tuple $\{c, d, w, b'\}$, where d is the distance from the search coordinate to the recommendation item, and b' is the aggregate of all bids b from content ratings matching the query.

Now, a user may view, or rate (vote on) a recommendation returned, as previously described.

Incentives

When a user votes on or contributes content that she likes, she can effectively use Gloe to build up personal bookmarks. Furthermore, Gloe can be used for social bookmarking, since a user can choose to only view content that her friends recommended. These uses of Gloe may incentivize a user to contribute and rate recommendations, since both she and her friends can directly benefit.

To give users an additional incentive to rate and contribute recommendations truthfully we pay users that contribute valuable recommendations. In economic terms, such payments can alleviate free-riding [22]. In particular, we pay each user who has contributed the most highly ranked u in any geographic area a within a time interval $[T_t, T_{t+d}]$ a bonus that increases her budget B. In the current system, d is set to a day, and there are about 33 thousand possible areas where users may earn bonuses. The mechanism was designed to reward users who predict popularity of Web content well and who contribute in areas where few ratings and recommendations exist. Even though there are other potential ways to reward users, we note that in our mechanism, only the user who contributed a rating on a new url or was the first to rate a url provided by the system is subject to a bonus. Moreover, the time interval only applies to the ranking whose url is the most popular the day the bonus is given, not the time the contributor made the initial rating.

The bonus rewards are also displayed on a global top list to show what content is popular and where, as a reputation and friend-finder mechanism. This potentially further incentivizes users to contribute, since contribution exhibits a strong positive dependence on attention in crowdsourcing [16]. We also list the top contributors based on aggregate bids within a channel and a geographic area, a, where a user's search originates from.

4.2 Design

We have designed and built a service, based on the model just discussed, that efficiently serves geographically as well as social-network-filtered recommendations on folksonomy tagged Web content. A number of mobile clients have also been implemented to evaluate the service.

The key design points in the Gloe service are i) aggressive partitioning of the data based on geography, while offering rich query capabilities within each partition, ii) social network use for query-by-query filtering, iii) easy and efficient mobile device and Web access. Below we discuss the main design decisions.A more comprehensive description of the design is outside the scope of this paper.

To be able to serve rich SQL queries efficiently with low latencies required for mobile client usage, we split the recommendation data in geographically partitioned databases, here called *shards*. Each partition represents a geographic region approximately 100 by 100 miles large, based on the first three hash characters rendered by the Geohash [12] algorithm. Currently we host about 7 million recommendations, in about 12 thousand regions. This partitioning makes expensive SQL radius queries fast even with large amounts of data. Partitions may be hosted on a single multi-core machine or be distributed across a cluster of machines. In either case throughput may improve dramatically, and scaling up based on demand is easy. We also improve the performance of our system by mainly indexing Web content, and meta-data as opposed to hosting and serving the content.

The Gloe service integrates with the Facebook authentication mechanism to allow filtering of recommendations based on your existing social network. We have also implemented an open HTTP/JSON protocol [13] that can be easily accessed from many mobile or Web based platforms. To date we have mobile clients for Android, BlackBerry, iPhone, and WebOS as well as a general purpose HTML5/AJAX based Web client that works in all the major Web browsers. Screenshots of the Web and mobile clients are shown in Figure 1 and Figure 2.

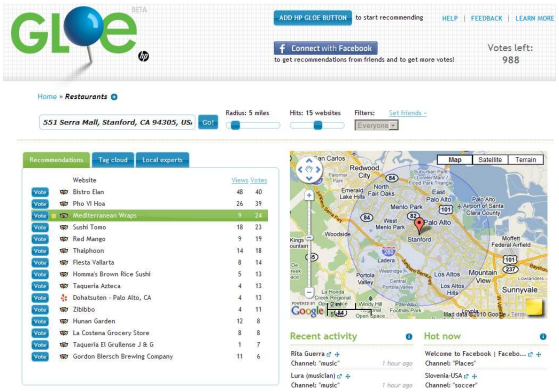

Figure 1: *The Web interface at hpgloe.com displaying channel menu (top); recommendations (left); map, recent and trending recommendations (right).*

5. SYSTEM EVALUATION

In this section we evaluate Gloe including its rating ratio, query success rate, performance, and geographic coverage based on traces from about 5 months of usage.

[12] http://wikipedia.org/wiki/Geohash
[13] http://hpgloe.com/api

Figure 2: *The Android, iPhone, BlackBerry, and WebOS (left to right) clients showing top channels and recommendations.*

Table 1: Recommendation Click or Rating Success (Rating Success in parenthesis)

Channel	Success	Path	IPs	Sessions
Overall	0.59(0.52)	7.04(7.96)	1152(1132)	2211(1875)
Travel	0.87(0.83)	7.22(7.29)	50(45)	67(52)
Cities	0.70(0.64)	3.73(3.94)	31(31)	33(33)
Photos	0.66(0.64)	6.08(6.43)	356(342)	478(446)

5.1 Rating to click ratio

First we studied whether our system improved on the rating ratio as defined in Section 3. We looked at unique Web pages [14] across all of our regions in our system (366 URLs) and computed a rating ratio for all of them. We found $\bar{\mu} = -0.135 \pm .049$ (see Equation 1 in Section 3). This value is significantly higher than the ratios studied in Section 3. It is on average about 107, 34 and 3 ($10^{2.165-.135}$, etc) times higher than the ratios for YouTube, Android Market and Digg respectively. This result is significant on a 5% level with a one-tailed z-test.

5.2 Recommendation success

To evaluate how successful our recommendations were we studied a five-month trace (Feb-July 2010) containing $507,907$ records from the Web server log where we reconstruct user sessions based on activity from a total of $13,059$ IP addresses. A user session is defined as activity from the same IP that starts by a call to our recommendation retrieval API and ends either with our click or rating API (success) or by timing out after 30 minutes of inactivity (failure), a limit commonly used [4] in these kind of studies. We wanted to know the success rates and session lengths for clicks and ratings across different channels, which represent a proxy for a keyword search in our system. Table 1 shows the statistics for the case where a session may end with both a rating and a click. Given the high rate of ratings we also show the data for the case where only a rating is considered to end the session successfully in parentheses. The most interesting part of this data is that the session query success rate only drops from 0.59 to 0.52 and the session length only goes up from 7.04 to 7.96 when only ratings not clicks are considered a success, which again supports the hypothesis that our system encourages ratings.

5.3 Performance

To evaluate the performance and scalability of the service we pre-loaded the system with about 6 million recommendations from crawls of Wikipedia, Wikitravel, Panoramio and other on-line portals with geotagged information. For the experiment we deployed one server on a 1-Core Amazon EC2 *Small Instance* virtual machine, and another server on a 2-Core EC2 *High-CPU Medium Instance* virtual machine. Both servers ran Fedora 8 with 1.7 GB of RAM.

We studied the most expensive query (radius search) executed for each of the 83,000 largest cities in the world, which accounts for all cities with a population greater than $1,000$. We measured the response time and throughput for 1-3 concurrent streams of requests, all obtaining the top 50 recommendations within 30 miles of the current location across all channels. Both measures are total round-trip times from a single client machine, including network latency (within a datacenter), Web server RPC and database processing.

Figure 3 shows the result. We can see that the median response time is around 30ms and the throughput scales to over 40 requests per second on the 2-Core machine. We also note that we achieve close to twice the throughput on the 2-Core machine compared to the 1-Core machine, which shows that the system scales well.

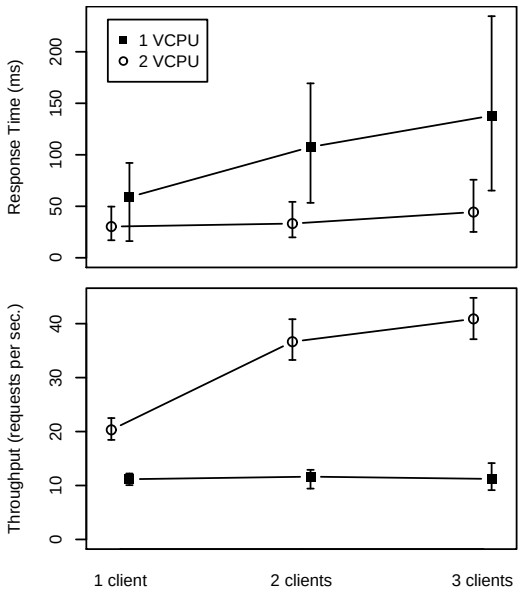

Figure 3: Throughput and response time of radius queries for a 1 and 2-Core server with 1-3 concurrent streams. The error bars represent the 1st and 3rd quartile, and the line represents the medians.

5.4 Coverage

To see if we have any biases in coverage across our 12,000 regional databases we graphed a recommendation distribution in Figure 4 and we produced a 256 color heatmap (see Figure 5) of the world, where each pixel is colored according to a scale from black (fewest) to white (most) depending on the number of recommendations in that location. From the graph in Figure 4 we can see that there is a fairly even distribution of recommendations across shards.

Gloe currently has 3427 users (297 Facebook users, 103 Gloe-registered users, 3027 anonymous users), 261 notes, 6.9 million recommendations on about 6.7 million URLs with $8,559$ tags. We have recorded $3,678$ recommendations and $9,478$ clicks on recommendations (all data from Dec 2009-July 2010). Our users come from 75 different countries, the top ones being the US, India, the UK, and Canada.

[14] that had been either clicked on or rated more than 3 times, with at least one click and one rating

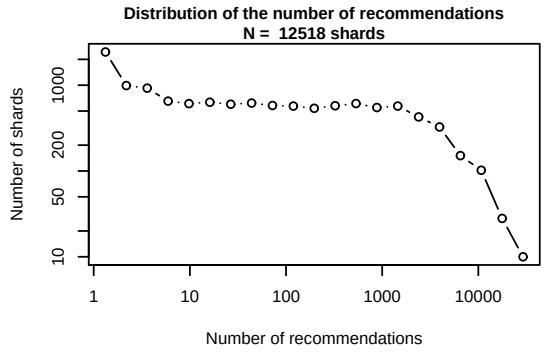

Figure 4: Distribution of recommendations, clicks and ratings with 3-level shard partitioning

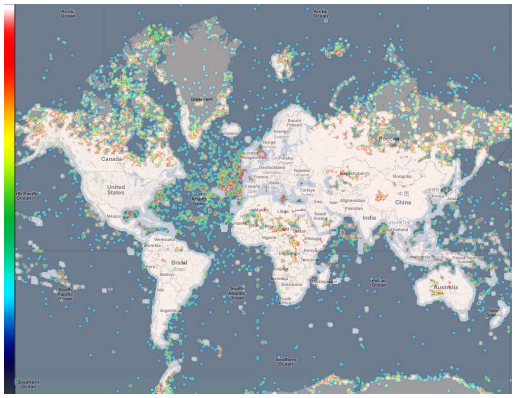

Figure 5: World HeatMap of recommendations with a color scale on the left from least (bottom) to most (top) recommended.

6. MECHANISM EVALUATION

Gloe allows a user to assign arbitrary amounts of her budget to various URLs, thus recommending the corresponding content. In this section we discuss a series of experiments on Amazon Mechanical Turk[15] that evaluate this mechanism by comparing it to a widely used voting mechanism as well as variations with and without incentive considerations.

We consider the following two mechanisms:

- **5-star mechanism:** This mechanism asks users to rate on a scale from 1 to 5.
- **100-credit budget mechanism:** This mechanism assigns a budget of 100 credits to each user, and asks a user to allocate this credit among alternatives. A user does not have to spend all her budget.

The 5-star mechanism is widely used by recommendation websites (such as Yelp). On the other hand, the 100 credit mechanism, an abstraction of Gloe's mechanism, allows a finer grained budget distribution and thus gives users more freedom over how to rate.

6.1 Mechanical Turk experiments

The purpose of these experiments is to evaluate the influence of incentives for rating Web pages as well as the differences in extracting knowledge from users with the 5-star mechanism compared to the 100-credit budget mechanism. The incentive and budget mechanisms tested here are abstractions of the mechanisms presented in Section 4.1 to fit the Mechanical Turk setup. For example, more than one user may receive a bonus for a particular location in the experiments, because we want to have enough predictions for the same location to make more sensible comparisons between individual and average ratings.

For each of 6 different locations, we selected 10 Web pages that could be useful either for a person living in that location or for someone visiting (to illustrate that Gloe can be used in both settings), and ran the following 5 experiments on Mechanical Turk.

(1) 5-star mechanism without incentives (**SN**): Users are asked to rate the 10 web pages on a scale 1-5.

(2) 100-credit budget mechanism without incentives (**BN**): Users are asked to rate the same 10 web pages by spending 100 credits across them in the same area as A.

(3) 5-star mechanism with incentives (**SI**): Like SN but with the additional instruction that they will receive a bonus payment paid in a raffle proportional to how well they predict the popularity of the 10 sites.[16]

(4) 100-credit budget mechanism with incentives (**BI**): Like BN but with the additional instruction that they will receive a bonus payment paid in a raffle proportional to how well they predict the popularity of the 10 sites.

(5) 100-credit budget mechanism with incentives on Gloe (**BG**): Users are asked to rate the 10 pages directly on Gloe. Each user is allowed to spend up to 100 credits. Users are again told that they will receive a bonus payment paid in a raffle proportional to how well they predict the popularity of the 10 sites.

We thus run 30 different experiments (HITS in the Mechanical Turk Terminology). For each HIT, we had the participation of 50 users. In total, there were 500 unique users.

We selected the following 6 locations: Chicago, IL; Palo Alto, CA; Mumbai, India; Bangalore, India; Paris, France; and Athens, Greece. We selected two cities in the United States and two cities in India, because the majority of workers on Mechanical Turk reside in these countries [17]. On the other hand, we selected Paris and Athens, because they are popular tourist destinations.

6.2 Budgets and incentives

We next compare the results of the experiments using two metrics: the Kendall τ rank correlation coefficient [21] and the root mean squared error (RMSE). The τ metric is used to determine how well the mechanism ranks the items, whereas the RMSE metric is used to measure how well the absolute aggregate value is predicted. Both the rank and the absolute values are visible to users in Gloe, and we thus want to optimize their accuracy. Accuracy here is defined as how far off the average or aggregate values individual users predict the popularity of a Web page.[17]

Amazon Mechanical Turk users only spend a limited amount of time for each experiment. In order to have the users spend this time on thinking about the popularity of pages in the given location, and not on calculating whether the points they allocated sum to 100, we allowed users to assign a total amount that exceeds 100 in experiments BN and BI. However, users were told that if the total amount exceeded 100, then the values they entered would be normalized. We normalize the values for BN and BI accordingly.

To compare the qualities of recommendations of the 100-credit

[15]https://www.mturk.com

[16]The results from the SI experiment are only used in Section 6.3.

[17]We could have potentially used different metrics to evaluate the accuracy or quality of a recommendation, such as looking at how well the intrinsic quality of content is predicted. However, it is hard to evaluate such metrics, since the intrinsic quality is not known to us. At any rate, for a recommendation system like Gloe a good recommendation is a recommendation that other users find valuable; thus it is important to predict what other users will think about specific content.

Table 2: Mean Value of Kendall's Coefficient (τ) and Root Mean Squared Error (RMSE)

	SN	BN		BI		BG
	τ	τ	RMSE	τ	RMSE	τ
Athens	0.14	0.14	5.38	0.08	4.48	0.27
Bangalore	0.39	0.47	5.30	0.43	5.32	0.48
Chicago	0.34	0.36	5.46	0.34	5.07	0.28
Mumbai	0.24	0.27	5.09	0.25	4.62	0.32
Palo Alto	0.49	0.51	4.89	0.52	4.79	0.45
Paris	0.29	0.31	5.68	0.25	4.96	0.32

budget mechanism and the 5-star mechanism, we first use the Kendall τ rank correlation coefficient [21], a measure of rank correlation. Kendall's coefficient lies between -1 and 1, and increasing values imply increasing agreement between the rankings. If the agreement between the two rankings is perfect (i.e., the two rankings are the same) the coefficient has value 1. If the disagreement between the two rankings is perfect (i.e., one ranking is the reverse of the other) the coefficient has value -1.

The τ column in Table 2 shows the average value of Kendall's coefficient when comparing a user's entry with the sum of all users' entries in a given experiment. A larger value is better. We observe that BN performs better than SN with respect to this metric, indicating that *the 100-credit budget mechanism results in better recommendations than the 5-star mechanism when users are not given monetary incentives.* A paired t-test shows that this is significant at a 5% significance level.

Having shown that budgets improve the performance without incentives, we next investigate whether incentives further improve the performance. We compare the experiments through the root mean squared error (RMSE).[18] In particular, for each experiment we compute the average RMSE of the difference between the ratings of a user and the average ratings of all users. A small RMSE indicates that users gave good recommendations, since their votes were well aligned with the votes of other users. The RMSE column in Table 2 shows the RMSEs for the BI and BN experiments. We observe that experiment BI exhibits a smaller RMSE than experiment BN (for most locations) implying that incentives improve the quality of recommendations when each user can spend 100 credits. A paired t-test shows significance at the 5% level. Thus, Table 2 also demonstrates that *users give better recommendations when they are incentivized to do so under the 100-credit budget mechanism.* This, in combination with the fact that BN performs significantly better than SN with respect to Kendall's coefficient, shows that both the budget and the incentive mechanisms could improve the quality of the aggregate ratings.

Finally, the τ column in Table 2 shows that the performance of BI and BG is approximately the same, validating Gloe's mechanism. In particular, Kendall's coefficient is slightly bigger for BG than BI, but the difference is not statistically significant.

6.3 Self-selection and quality

Since we allow Mechanical Turk workers to take any subset of our experiments, each worker could take up to 18 bonus-based surveys (SI, BI, and BG). In this section we evaluate the self-selection process to determine what positive or negative effects bonuses had on continued participation. The reason for studying this behavior is to see whether incentives could be used to filter out a higher quality subset of users to increase the overall quality of contributions over time.

[18]Kendall's coefficient does not give significant results for the comparison of BN and BI. On the other hand, RMSE is not a good metric to compare SN with BN, because of the different range of ratings.

Table 3: Self Comparison Bonus Feedback Effect on Participation

Surveys	Pos Signal	Neg Signal	Users
2	0.47	0.72	86
3	0.52	0.70	22
4	0.59	0.69	31
5	0.63	0.58	13
6	0.54	0.60	7

The general evaluation method is to look at the bonus performance of users clustered by the number of surveys they take. We then compare the bonus dynamics for each user in two tests. In the positive signal test, if a user gets a bonus greater than or equal to the last bonus received we say that a *positive signal* has been sent to the user. Conversely, in the negative signal test, if a user gets a bonus less than or equal to the last bonus received we say that a *negative signal* has been sent to the user. This setup was used because many times the same bonus is received and we did not want to bias the results based on which bucket (positive or negative) we assigned these bonuses to. Furthermore, we limit the clusters to 6 surveys taken since higher number of surveys would result in clusters that would be too biased towards individual performances.

We then compare the fraction of positive and negative signals for each cluster of users. If the bonus incentives work we would expect to see an increasing fraction of positive signals and a decreasing fraction of negative signals for clusters with more surveys taken.

In Table 3 we can see that the expected trends appear for clusters of 2,3,4, and 5 surveys but then break. This could be due to the fact that we used only 3 surveys for each location in our experiments and it may be difficult to predict well for locations the survey taker is not familiar with.

7. CONCLUSION

In this paper we presented the design, implementation and evaluation of a new geotagging service, Gloe, that makes it easy to find, rate and recommend arbitrary on-line content in a mobile setting. The Gloe rating mechanism consists of a global budget which users can utilize for making local recommendations. The fact that a global budget can induce the right behavior is by no means obvious, for it could in principle incentivize frivolous recommendations of places and votes about which users have little or no expertise. And yet our Mechanical Turk experiments show that in spite of its diffuse geographical nature, (6 cities across the world) such global budget is effective at extracting a higher quality aggregate ranking than without the budget. The budgets used in the experiments are global in the sense that all users regardless of where they reside are given the same budget.

High rating participation is crucial for the quality of the results when statistically aggregating crowdsourced opinions. It is hence promising that we were also able to show that the rating to click ratio for the live Gloe system that implements the rating mechanism was 3-107 times higher than the ratio for similar systems.

We note that our recommendation mechanism could be applied beyond local Web content ratings in any situation where a trade-off needs to be made between low-effort opinion sharing and high-quality contributions, such as reader review of editorials, evaluation

of customer support service, rating of print kiosks and voting on innovation ideas.

Our future work will focus on designing more sophisticated economic mechanisms to allow users with knowledge about overrated or underrated content to earn money on arbitrage like on the stock market. Furthermore, we are interested in studying how popularity and novelty can be traded off in this geographic setting so as to maximize click-through rates. A project is also underway to apply the Gloe mechanisms to collect volunteer annotations for remote parts of Africa in applications including humanitarian aid.

Acknowledgments

We thank Yarun Luon for developing the BlackBerry client and providing useful comments; and Gabor Szabo for providing the data from YouTube.

8. REFERENCES

[1] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.*, 23(1):103–145, 2005.

[2] E. Amitay, N. Har'El, R. Sivan, and A. Soffer. Web-a-where: geotagging web content. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 273–280, Sheffield, United Kingdom, 2004. ACM.

[3] S. Asadi, X. Zhou, and G. Yang. Using local popularity of web resources for geo-ranking of search engine results. *World Wide Web*, 12(2):149–170, 2009.

[4] R. Baraglia, F. Cacheda, V. Carneiro, D. Fernandez, V. Formoso, R. Perego, and F. Silvestri. Search shortcuts: a new approach to the recommendation of queries. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 77–84, New York, NY, USA, 2009. ACM.

[5] R. Bhattacharjee and A. Goel. Algorithms and incentives for robust ranking. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 425–433, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[6] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems*, pages 107–117, 1998.

[7] Y.-Y. Chen, T. Suel, and A. Markowetz. Efficient query processing in geographic web search engines. In *SIGMOD '06: Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 277–288, New York, NY, USA, 2006. ACM.

[8] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1):242–259, March 2007.

[9] R. Forsythe, F. Nelson, G. Neumann, and J. Wright. The iowa presidential stock market: A field experiment. *Research in experimental economics*, 4:1–43, 1991.

[10] J. Freyne, M. Jacovi, I. Guy, and W. Geyer. Increasing engagement through early recommender intervention. In *Proceedings of the third ACM conference on Recommender systems*, pages 85–92. ACM, 2009.

[11] A. Goodman. *Winning Results with Google AdWords, Second Edition*. McGraw-Hill Osborne Media, 2008.

[12] R. V. Guha. Programmable search engine, February 2007. US Patent Application 202423:11.

[13] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev, and S. Ofek-Koifman. Personalized recommendation of social software items based on social relations. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 53–60, New York, NY, USA, 2009. ACM.

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

[15] B. A. Huberman. The social mind. In J.-P. Changeux and J. Chavaillon, editors, *Origins of the Human Brain*, pages 250–261. Oxford University Press, 2002.

[16] B. A. Huberman, D. M. Romero, and F. Wu. Crowdsourcing, attention and productivity. *Journal of Information Science*, 35(6):758–765, 2009.

[17] P. Ipeirotis. Demographics of Mechanical Turk. *CeDER Working Papers*, 2010.

[18] A. Josang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2):618–644, Mar. 2007.

[19] M. Kamvar and S. Baluja. A large scale study of wireless search behavior: Google mobile search. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 701–709, New York, NY, USA, 2006. ACM.

[20] M. Kamvar and S. Baluja. The role of context in query input: using contextual signals to complete queries on mobile devices. In *MobileHCI '07: Proceedings of the 9th international conference on Human computer interaction with mobile devices and services*, pages 405–412, New York, NY, USA, 2007. ACM.

[21] M. Kendall. A new measure of rank correlation. *Biometrika*, 30(1–2):81–89, 1938.

[22] A. Mas-Colell, M. D. Whinston, and J. R. Green. *Microeconomic Theory*. Oxford University Press, Oxford, United Kingdom, 1995.

[23] D. M. Pennock, S. Lawrence, C. L. Giles, and F. A. Nielsen. The real power of artificial markets. *Science*, 291:987–988, February 2001.

[24] H. Rheingold. *Smart Mobs: The Next Social Revolution*. Basic Books, 2002.

[25] J. Schiller and A. Voisard. *Location Based Services*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[26] J. Surowiecki. *The Wisdom of Crowds*. Anchor, 2005.

[27] G. Szabo and B. A. Huberman. Predicting the popularity of online content. *Communications of the ACM*, August 2010. to appear.

[28] F. Wu and B. A. Huberman. Novelty and collective attention. *Proc. Natl. Acad. Sci (USA)*, 104:17599–17601, 2007.

[29] F. Wu, D. M. Wilkinson, and B. A. Huberman. Feedback loops of attention in peer production. In *CSE '09: Proceedings of the 2009 International Conference on Computational Science and Engineering*, pages 409–415, Washington, DC, USA, 2009. IEEE Computer Society.

[30] Y. Zhen, W. Li, and D. Yeung. TagiCoFi: tag informed collaborative filtering. In *Proceedings of the third ACM conference on Recommender systems*, pages 69–76, New York, New York, USA, 2009. ACM.