Proceedings of the First AAAI Conference on Human Computation and Crowdsourcing

Improving Your Chances: Boosting Citizen Science Discovery

Yexiang Xue and **Bistra Dilkina** and **Theodoros Damoulas**
Cornell University, Ithaca, NY
{yexiang, bistra, damoulas}@cs.cornell.edu

Daniel Fink
Cornell Lab of Ornithology, Ithaca, NY
df36@cornell.edu

Carla P. Gomes
Cornell University, Ithaca, NY
gomes@cs.cornell.edu

Steve Kelling
Cornell Lab of Ornithology, Ithaca, NY
stk2@cornell.edu

Abstract

Citizen scientists are playing an increasing role in helping collect, process, and/or analyze data used to study a variety of scientific phenomena. We address the problem of identifying tasks that are rewarding to the citizen scientists, which results in greater participation, leading to more data and better models. We apply our methodology to *eBird*, whose participants are avid birders interested in observing different species while contributing to science. In order to improve the birders' chances of meeting their goals, we consider the following *probabilistic maximum coverage* problem: Given a set of locations, select a subset of size k, such that the birders maximize the expected number of observed species by visiting such locations. We also consider a secondary objective that gives preference to birding sites not previously visited. We consider two variants of the *probabilistic maximum coverage* problem, provide a theoretical analysis, describe several algorithms with provable approximation guarantees, as well as heuristic approaches, and provide empirical results using *eBird* data. Our algorithms are fast and provide high quality recommendations.

1 Introduction

The advancements in Information Technology, such as the World Wide Web, mobile devices, and social networking technology, have provided new opportunities for large-scale citizen science programs (Bonney et al. 2009). Citizen science engages the public in collecting, processing and/or analyzing data, with the goal of contributing to scientific research. A large number of successful citizen science applications have been developed in recent years, with online citizen science communities contributing to a variety of projects across different disciplines. For example in astronomy, citizen scientists classify galaxies in Galaxy Zoo (Lintott et al. 2008) and search for new exoplanets, i.e., Earth-like planets beyond our solar system, in Planet Hunters (Schwamb et al. 2012).In biology, citizen scientists contribute to bird and arthropod research using *eBird* (Sullivan et al. 2009) and BugGuide (Bartlett 2011). In environmental studies, citizen scientists help monitor coral bleaching trends (Marshall, Kleine, and Dean 2012).

Our research is motivated by our collaboration with the Cornell Lab of Ornithology. The Cornell Lab of Ornithology has developed a variety of citizen science projects concerning bird conservation, each designed to inform specific scientific questions, while engaging the public in science [1]. For example, *eBird* enlists bird watchers to identify bird species, a task that only humans are able to reliably perform, given current technology. In *eBird*, bird watchers report their observations to a centralized database via online checklists that include detailed information about the observed birds, such as the species name, number of individuals, gender, time and location of the observation. To date more than 141,000 individuals have volunteered more than 9 million hours and collected over 125 million bird observations. Since 2006, *eBird* data have been used to study a variety of scientific questions, from highlighting the importance of public lands in conservation to studies of evolution and climate change (Kelling et al. 2012).

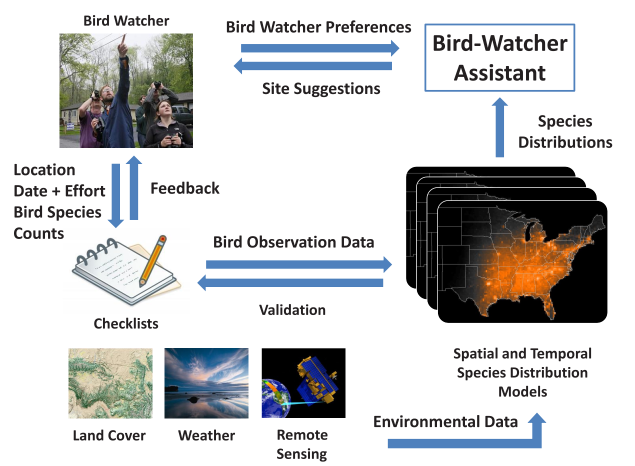

Figure 1: Bird-Watcher Assistant recommends birding sites to improve the birders' chances of seeing a set of diverse species, combining birders' information with species distribution information, inferred from predictive spatial-temporal models that integrate bird observational data, submitted by the birders, with environmental data.

The overall scientific goal of citizen science projects often involves the study, understanding, and characterization

[1] http://www.cornellcitizenscience.org

of phenomena that occur across different spatial and temporal regions: citizen scientists play a key role in helping gather, process, and/or analyze data used for the development of predictive models of such phenomena. Citizen science projects face several challenges in order to ensure (1) a high level of participation and engagement of citizen scientists, (2) a reasonable distribution of the citizen scientists' contributions (e.g., geographically and throughout the year) and (3) high quality of the citizen scientists' contributions. In this paper we address some of the issues concerning the challenges (1) and (2). We illustrate our research and methodology using *eBird*, but our results can be generalized to other citizen science projects.

eBird's approach to stimulate participation and engagement of citizen scientists is to develop clear rules of participation and incentives that appeal to the birding community (Wood et al. 2011). *eBird* provides several record-keeping, exploration, and visualization tools that nurture and reward participation. The success of *eBird*, with exponential data growth since 2006, is in part due to the fact that it appeals to the competitiveness of the participants, providing a variety of tools that allow participants to determine their relative status compared to other participants (such as numbers of species seen) and by geographical regions (such as checklists submitted per state and province).

To further boost participation and scientific discovery in *eBird*, we are developing a set of tools for recommending interesting birding sites, encapsulated in an application we call *Bird-Watcher Assistant*. Figure 1 provides a high-level view of *Bird-Watcher Assistant* and Figure 2 shows a snapshot of the birding sites suggested by the system, using hotspots voted by birders. *Bird-Watcher Assistant* uses information from species distribution models, which predict species occurrence at a given location and date based on the associations between current *eBird* observations and local environmental data. These species distribution models inform the selection of the most desirable or useful new tasks for the citizen participants. A related process is active learning, in which one seeks to select the set of unlabeled data points that when added to the labeled training data would have the most significant impact on the fitted predictive model. In the context of citizen science, however, one cannot simply maximize informativeness of the tasks but has to take into account the interests of the citizen scientists to maintain high participation rates. In *eBird*, participants are avid birders who are interested in contributing to science, but also enjoy seeing a diverse sets of species. Designing tasks that are rewarding to the citizen scientists results in greater participation, which in turn results in more data, better models, hence to better designed tasks.

In order to improve the birders' chances of seeing a diverse set of species, we consider the following problem: *Given a set of locations, select a subset of size k, such that the birders maximize the expected number of observed species by visiting such locations.* We consider two variants of this problem: (1) a local scale variant, in which we are choosing among birding sites that are within a given region, for example when planning a birding trip within a county, and (2) a large scale variant, in which we want to choose a sub-region (with a given radius), from a given larger region, from which we want to choose the set of locations to visit. For example, birders might want to fly to Colombia and visit a sub-set of birding sites within a sub-region of Colombia. We formalize the first problem as the *probabilistic maximum coverage* problem, and the second as *probabilistic maximum coverage with locality constraints*.

We note that in addition to the primary objective of maximizing the expected number of observed species we also consider a secondary objective that gives preference to birding sites not previously visited, when in the presence of multiple solutions with a comparable number of expected species. This secondary objective helps expand the spatial coverage of *eBird* by promoting new birding sites, typically in less populated areas. Observations made at the *Bird-Watcher Assistant* recommended sites will help mitigate the spatial bias in *eBird* where observations are concentrated toward regions with high human density.

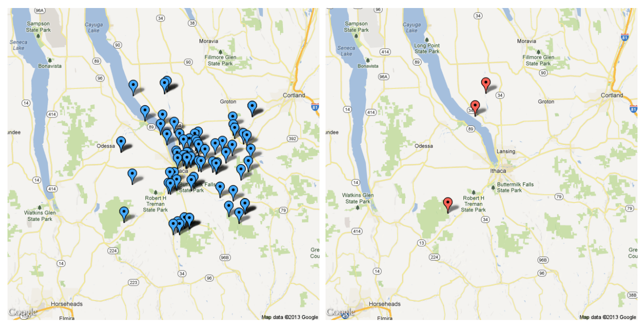

Figure 2: (Left) Interesting birding sites in a county; (Right) A subset of three sites recommended by *Bird-Watcher Assistant*: a forest, a lake-side, and a grassland site.

We show that the problem of *probabilistic maximum coverage* can be formulated as maximizing a submodular function, subject to cardinality constraints. While we show that the problem is NP-hard, we use the classical $(1-1/e)$ approximation algorithm (Nemhauser, Wolsey, and Fisher 1978), and compare our results with the sets of locations recommended by human experts. We then show that the problem of *probabilistic maximum coverage with locality constraints* can also be encoded as optimizing a submodular function, but subject to both cardinality and locality constraints, specified by a given radius. To our knowledge, the most similar problem studied previously concerns submodular optimization subject to a path length constraint (Chekuri and Pál 2005; Singh et al. 2009). The state-of-the-art for that problem is a quasi-polynomial algorithm with a logarithmic approximation bound. In contrast, we are able to prove that our problem, with radius locality constraints, admits a strongly polynomial $(1-1/e)$ approximation bound. This algorithm makes a quadratic number of calls to the classic submodular greedy algorithm, and in practice, when the number of locations to choose from is large, it is still not practical. To address this issue, we propose a bi-criteria approximation algorithm that relaxes the locality constraint, but makes only a linear number of submodular optimization calls. We also propose a local search based sampling method, without optimality guarantees.

We evaluate the performance of the proposed algorithms in the context of *eBird*. At the local scale, we consider Tompkins County, NY, the home of the Cornell Lab of Ornithology and *eBird*. To test the performance of *Bird-Watcher Assistant*, we compared locations recommended by our model to locations recommended by a set of expert birders. Qualitatively, the locations suggested by our model were judged to be of quality by the domain experts. Quantitatively, the locations suggested by our model achieve higher expected numbers of species than the locations suggested by the experts. The *Bird-Watcher Assistant* locations systematically covered the three most important habitat types for birds, while promoting increased spatial coverage of the county. At a larger scale, we consider planning birding trips across multiple states, spanning more than 70,000 potential locations, revealing that in practice our local search based sampling method performs very close to the approximation algorithm but with a much better runtime. Overall our algorithms are remarkably fast and provide high quality birding site recommendations.

In the rest of the paper, we formulate the two variants of the *probabilistic maximum coverage* problem and provide a theoretical analysis, describe several algorithms with provable approximation guarantees, as well as heuristic approaches, and provide empirical results.

2 Local Scale Problem: Probabilistic Maximum Coverage

Birders are often interested to know: what are the 5 most interesting places to go birding in a given area? Typically birders can visit any interesting location within a relatively small region such as a county during a day or a weekend trip, and hence do not care about the distance between locations within such a region. However, birders might be limited to visiting at most a given number of places, due to both time and resource constraints. Although there are many reasons one can consider a location or a set of locations "interesting", most birders are concerned with maximizing their chances of observing different species. Avid birders, for example, participate in online birding contests such as the eBird Top100 lists,[2] where birders are ranked by the number of different species they have observed within a given county, state, or region. To support birders in planning day trips at a local scale, we consider the following problem: *what is the set of k locations within a given region that when visited maximizes the expected number of species observed?*

Formally, suppose a birder has a list of m species that he/she considers interesting. Let $P = \{1, 2, \ldots, n\}$ be the set of all candidate locations within the region of interest. We assume the existence of prior models of species distributions. For a given time of the year, let p_{ij} be the probability of observing species $i \in \{1..m\}$ at location $j \in \{1..n\}$. Note this should not be an important limitation. In general, when such a prior model does not exist in the beginning, a uniform prior can be used which would recommend uniformly spread locations. Let $f(S)$ be the expected number of species seen by visiting all places in S, $S \subseteq P$. Let $X_i(S)$ be a binary random variable where $X_i(S) = 1$ if and only if we observe species i when visiting all places in S. We assume that the number of detections of a single species seen by visiting all places in S follows an inhomogeneous Poisson process (Diggle 2003) where the intensity of detections vary with local environmental features. Thus, given the number of detections in S, these detections form an independent random sample and the probability of observing species i by visiting S is 1 minus the probability of not observing the species at each of the location in S, i.e.: $\Pr(X_i(S) = 1) = 1 - \prod_{j \in S}(1 - p_{ij})$. We can now define our problem: PROBABILISTIC MAXIMUM COVERAGE (short name: `k-BestPlaces`):

$$\text{maximize } f(S), \text{subject to } |S| \leq k.$$

where the total expected number of observed species by visiting S is:

$$f(S) = E[\sum_{i=1}^{m} X_i(S)] = \sum_{i=1}^{m} \left(1 - \prod_{j \in S}(1 - p_{ij})\right).$$

Theorem 2.1. *$f(S)$ is submodular and monotone.*

Theorem 2.2. PROBABILISTIC MAXIMUM COVERAGE $\in$ NP-COMPLETE.

Because $f(S)$ is a special case of a weighted coverage function as defined in (Călinescu et al. 2011), we can show it is submodular and monotone. Maximizing a general weighted coverage function subject to cardinality constraints is NP-hard; we show that it is still NP-hard if the function takes the special form of $f(S)$. The proofs of Theorem 2.1 and Theorem 2.2 can be found in the appendix.

Based on the classical result from (Nemhauser, Wolsey, and Fisher 1978), Algorithm 1 is a `(1-1/e)` approximation algorithm that given an instance of `k-BestPlaces` runs in $O(nmk)$ time, where n is the number of locations and m is the number of species in the list.

Data: Point set $P = \{1, \ldots, n\}$, submodular function $f : S \rightarrow \mathbb{R}$, and k, $k \leq n$.
Result: Point set S, $S \subseteq P$, $|S| \leq k$.
$S \leftarrow \emptyset$;
for i *in* $1 \ldots k$ **do**
  $p \leftarrow \text{argmax}_{p \in P \setminus S} f(S \cup \{p\}) - f(S)$;
  $S \leftarrow S \cup \{p\}$;
end
return S

Algorithm 1: `(1-1/e)` approximation algorithm for the `k-BestPlaces` problem.

3 Large-Scale Problem: Probabilistic Maximum Coverage with Locality Constraints

Consider a birder who lives in upstate New York. He would like to plan a birding trip going anywhere within 300 miles

[2] http://ebird.org/content/ebird/about/about-the-ebird-top100

from his home. However, once he decides on one destination, he could visit at most 10 nearby places around that selected place due to the time constraints of the visit. This leads to another interesting extension of our problem. Formally, we define the problem: PROBABILISTIC MAXIMUM COVERAGE WITH LOCALITY CONSTRAINTS (short name: `(k,r)-BestPlaces`):

$$\text{maximize } f(S), \text{subject to } |S| \leq k \text{ and}$$
$$\text{all points in } S \text{ are covered by a circle of radius } r.$$

Here we consider the spatial coordinates of all candidate locations and use Euclidean distance.

`(k,r)-BestPlaces` is a submodular optimization problem subject to both cardinality and locality constraints. There have been several studies on maximizing submodular functions beyond cardinality constraints. (See e.g., (Călinescu et al. 2011)). To our knowledge, the most relevant research related to our problem is by (Chekuri and Pál 2005), who consider maximizing a submodular function subject to the constraint that all vertices in the set are linked by a path of at most a given length. They propose an algorithm with quasi-polynomial runtime and a logarithmic approximation bound. Unfortunately, this method does not scale to problems with hundreds or thousands of locations. Later, (Singh et al. 2009) improve the runtime of the algorithm of (Chekuri and Pál 2005) by applying spatial decomposition heuristics and by using branch and bound to speed up the search, but they loose the formal approximation bound. Note that the path-length constraint considered in these two papers is slightly different from ours, thus a direct comparison to their algorithms is not possible. However, the path-length constraint is indeed an interesting variant and we look forward to addressing it in future research.

We developed `EnumAllCircles`, a polynomial time `(1-1/e)` approximation algorithm for the `(k,r)-BestPlaces`. We show that to enumerate all subsets of points that meet the locality constraints, one only needs to enumerate all pairs of points within $2r$ distance, and for each point pair of this type to only consider the set of points covered by each of the two circles of radius r that pass through the point pair (see figure 3 (Left)). Then for each such set of points, we apply the greedy algorithm 1. The overall complexity is $O(n^2(d + n_0mk))$, where n is the number of points, n_0 is the maximal number of points within a circle of radius r, m is the number of species, and d is the time to find the set of points that are covered by a circle. For full details, please see the appendix. Although it has polynomial runtime, `EnumAllCircles` does not scale very well to real instances. For instance, there are typically tens of thousands of locations in the problem instances we consider. Running on these instances, `EnumAllCircles` needs to enumerate millions of circles, which requires hours to days of computation time.

We developed `EnumHexagonCircles`, an algorithm that is much faster than `EnumAllCircles`, enumerating only a linear number of circles at the expense of providing weaker approximation guarantees. In particular, `EnumHexagonCircles` is a $(\frac{1}{3}(1 - 1/e), 2r)$ bi-criteria approximation algorithm returning solutions that can violate the locality constraint by up to two times the required radius and with objective value within $\frac{1}{3}(1 - 1/e)$ of optimum.

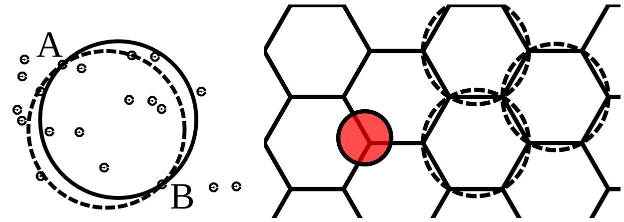

Figure 3: (Left) The solid and dash circles of radius r both pass through points A and B. (Right) A tessellation of regular hexagons of side length $2r$. Circles circumscribing hexagons are shown in dashed line. A circle of radius r can intersect with at most 3 hexagons (example is in red shade).

`EnumHexagonCircles` uses a tessellation of hexagons with side $2r$ across the space spanned by the input point set P (see Figure 3(Right)). `EnumHexagonCircles` only considers the circles circumscribing a hexagon in the tessellation and containing at least one point (See Algorithm 2). Because one point can be contained in at most three circles of this type, in the worst case the number of circles cannot exceed three times the number of points. Therefore, the number of calls on line 5 to Algorithm 1 is only linear in n. This results in an $O(n(d + n_0mk))$ overall complexity.

Data: Point set $P = \{1, \ldots, n\}$, f, k and r
Result: Point set S, $S \subseteq P$, $|S| \leq k$ and S is covered by a circle of radius $2r$.

1 $S_{best} \leftarrow \emptyset$;
2 Denote circle set $C_{tess}(r)$ to be all the circles circumscribing a fixed tessellation of regular hexagons of side length $2r$ across the space spanned by the input point set P;
3 **for** $C \in C_{tess}(r)$ *and* C *contains at least one point* **do**
4   extract points P_C covered by circle C;
5     $S \leftarrow$ Algorithm $1(P_C, f, k)$;
6   **if** $f(S) > f(S_{best})$ **then**
7         $S_{best} \leftarrow S$;
8   **end**
9 **end**

Algorithm 2: `EnumHexagonCircles`: a fast bi-criteria approximation algorithm for `(k,r)-BestPlaces`.

To prove the approximation bound, we first notice the following geometrical observation:

Proposition 3.1. *For a tessellation of regular hexagons of side length $2r$, as shown in Figure 3, any circle with radius r can intersect with at most 3 hexagons.*

Theorem 3.2. *Suppose the optimal value of problem* `(k,r)-BestPlaces` *is OPT. Algorithm* `EnumHexagonCircles` *returns a set of locations S_{best}, such that S_{best} is covered by a circle of radius $2r$ and $f(S_{best}) \geq \frac{1}{3}(1 - 1/e)OPT$.*

See the appendix for the proofs. We remark that `EnumHexagonCircles` represents a general class of algorithms harnessing the fact that a circle of a given radius can only intersect with a constant number of polygons in a tessellation, in the worst case. In this case, the optimal set of points can be split into at most a constant number of subsets, with each subset contained in a polygon of the tessellation. Similarly to Theorem 3.2, we can prove a constant approximation when we only apply submodular optimization for each polygon individually. As a second example, one can consider a tessellation of squares of side length $2r$, and design a similar algorithm that returns a set of points within a circle of radius $\sqrt{2}r$ and achieving $\frac{1}{4}(1-1/e)$ approximation bound.

4 Experiments

Local Scale Problem

We consider the study area of Tompkins County in New York State, where we have the highest density of observations in *eBird*, high spatio-temporal coverage, and available human expertise. We focus on $n = 165$ locations in the county that have been voted by birders as "hotspots" and a species list (number of species $m = 54$) that includes a diverse set of birds (resident, migrants, aquatic and forest species) native to the North East. Very common species such as the American Crow and the Black-capped Chickadee are excluded so that the resulting list is a representative, diverse portfolio of species that is of primary interest to birders[3].

The probability p_{ij}^t of observing species i at location j in month t is derived from spatiotemporal exploratory models (STEM) of this region for each month between January and June (Fink et al. 2010; Fink, Damoulas, and Dave 2013). The models utilize historical checklists from the *eBird* dataset, and employ a multi-scale strategy to model local and global spatiotemporal correlations. The resulting probabilities are on stratified random locations sampled from a grid of 3km x 3km pixels.

To quantify the performance of the approximation algorithm in practice, we implemented an exact brute-force algorithm, which enumerates all subsets of cardinality k and returns the best one. Because of the combinatorial nature of the problem, we are only able to compare with the brute-force algorithm for $k = 2$ and $k = 3$. The results show that the approximation algorithm performs considerably better than the guaranteed bound $(1 - 1/e) \approx 0.63$, recovering the exact solution on all instances except for a tiny loss ($< 0.18\%$) in February for $k = 3$. This empirical performance occurs because in smaller scales, such as the county-level, the probability distribution for a species is rather homogeneous for the same types of landscapes.

We compare both quantitatively and qualitatively the solutions obtained by our model to recommendations made by expert birders. We asked 10 experts from the Cornell Lab of Ornithology to list for each month the three best hotspots in Tompkins County that *collectively* maximize the number of species they expect to see. While other preferences such as aesthetics or convenient access might have indirectly affected experts' decision process, the experts were instructed to provide recommendations to maximize the performance in online birding contests, where birders are ranked by the number of different species they have observed within a given region. Figure 4 (Top) presents the expected number of species for each month based on the solution obtained by the greedy approximation algorithm and by the experts. [4] The algorithm outperforms the human experts across all months. This indicates that although the experts are very familiar with birding and the local hotspots, they cannot reason perfectly across many complementary locations and many diverse species. Hence, our tool will be useful to novices and experts alike, and will aid the scientific objectives of *eBird* by improving the information content of the citizens' contributions.

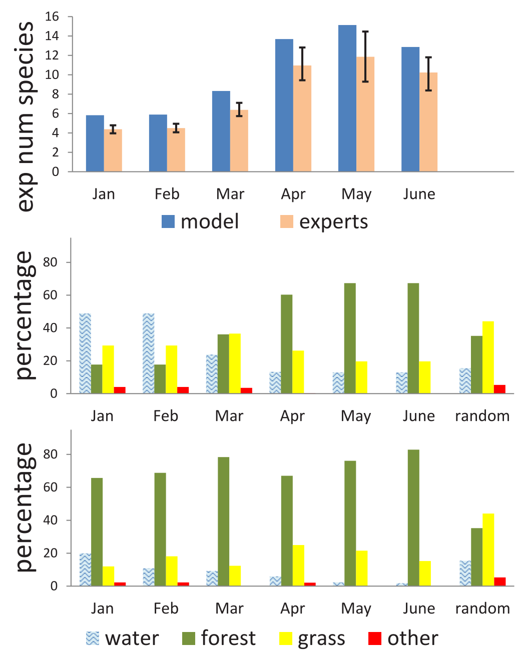

Figure 4: Results for Tompkins County with $k = 3$: (Top) Comparison of solutions returned by the greedy approximation algorithm with the experts' suggestions (showing the average among the 10 experts with the worst and best performance as error bars). (Middle) Landscape composition for the recommendations by the expert who achieves highest expected number of species. (Lower) Landscape composition for the solutions obtained by the model. The "random" column shows the landscape composition obtained by uniformly choosing three locations in Tompkins County.

[3]The dataset of hotspots for Tompkins County and the species list can be found online at www.cs.cornell.edu/~yexiang.

[4]We did not compare the number of species reported in historical checklists because in practice birding activities are highly biased towards famous places. As a result, the number of species reported in popular hotpots are significantly higher than those of nearby sites, even if they share the same environmental condition. From this point of view, it is unfair to make comparisons based on historical checklists, as it will improperly favor frequently visited locations.

To qualitatively compare the solutions between experts and algorithms, we study the distribution of land cover types across the sets of locations, as land cover is a significant factor in habitat suitability for different bird species. We estimate the landscape composition of a location by considering a 750-meter region around it, using the National Land Cover Database for the U.S. (Homer et al. 2007) and group the original categories into four classes: water, forest, grass, and other. The water class includes water and wetlands; the forest class includes deciduous, evergreen and mixed forests; the grass class includes shrub land, herbaceous, planted or cultivated land, open spaces and light intensity developed area; and the other class includes barren and developed land with mid to high intensity. For a set of locations, the aggregate landscape composition is computed by summing the area covered by each of the four land cover classes across all locations in the set and normalizing by the total area. Figure 4 (Middle) presents the results based on the recommendations of the best among the 10 experts (ranked by the expected number of species across the 6 months). Figure 4 (Bottom) presents the results obtained from the greedy approximation algorithm. We see that both the expert and model recommendations systematically cover the three important types of bird habitats: water, forest and grass. Both the expert and the model recommendations reveal similar trends, where water habitat is more preferred in winter months, while slightly more forest habitat is preferred towards the summer months. Ecologically these trends make sense as migrant birds, which include a lot of species with a woody habitat association, leave during the winter and return back during the summer months. Figure 5 shows maps of the solutions obtained by the greedy approximation algorithm for January and June.

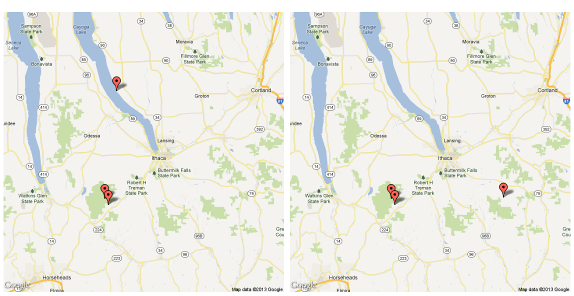

Figure 5: The solution for $k = 3$ suggested by the greedy approximation algorithm for January (Left) and June (Right).

Large Scale Problem

We consider an area along the East Coast of the U.S. (see Figure 7 (Right)), and focus on $n = 70,637$ stratified random locations specified in the STEM model output for that area. We use the same probability model and species list as in the local scale problem. Algorithms `EnumAllCircles` and `EnumHexagonCircles` described in section 3 are compared with the following algorithm variants:

- `SampleCirclesGreedy` This variant samples l circles of radius r uniformly at random; then it applies the submodular optimization for points in each circle, and selects the best answer among all the circles sampled. In our experimental setting, l is set to 10,000.
- `SampleCirclesRandom` This variant samples l circles of radius r uniformly at random; then it selects a random subset of size k among the points in each circle, and returns the best answer among all the circles sampled. It serves as a baseline to our algorithms. In our experimental setting, l is set to 10,000.

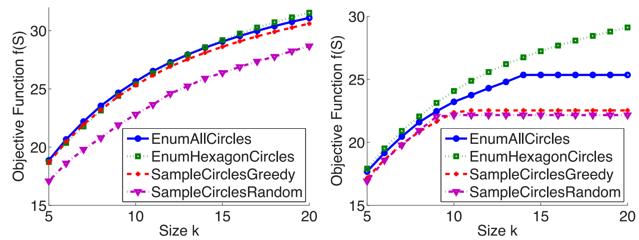

Figure 6: Comparison of the algorithms for the `(k,r)-BestPlaces` problem across varying k: (Left) June, $r = 20$ km; (Right) June, $r = 5$ km.

Figure 6 shows the performance of the different algorithms for $r = 5$ and $r = 20$ km for the month of June and for k varying from 5 to 20. Our results show that when r is reasonably large ($\geq$10 km, see Fig.6(Left)) `EnumHexagonCircles` and `SampleCirclesGreedy` return solutions of quality very close to `EnumAllCircles`. Since `EnumAllCircles` essentially enumerates all circles of interest, the only suboptimality of the obtained solution comes from using the greedy approximation algorithm for each point set covered by a circle. From our experiments on the local scale problem, we know that in practice the greedy approximation algorithm is likely to return solutions of quality much closer to optimal than the proven `(1-1/e)` bound, and hence also is `EnumAllCircles`. Unfortunately, in practice `EnumAllCircles` is computationally prohibitive for large number of locations. Table 1 provides a summary of the number of circles enumerated and the runtime for the algorithms for the month of June, $k = 20$ and $r = 20$ km. Note that switching to `EnumHexagonCircles` or `SampleCirclesGreedy` generates huge computational savings (see Table1). While `EnumHexagonCircles` might return solutions, which violate the locality constraints, `SampleCirclesGreedy` returns feasible solutions, which are of comparable quality. Hence, while lacking formal optimality guarantees, in practice `SampleCirclesGreedy` is both computationally efficient and accurate for large scale problems with larger r.

We note that it becomes harder to find good quality solutions when r is relatively small. For example, we experiment on a special case when $r = 5$ km as shown in Figure 6(Right). In this case `EnumHexagonCircles` likely overestimates the objective function, while relaxing the locality constraint to circles of radius $2r$. On the other hand, `SampleCirclesGreedy` and `SampleCirclesRandom` saturate early on and cannot find good solutions for large values of k. Intuitively, a smaller radius makes the problem harder, as there are fewer circles that cover a particular set of points. Thus, to find the optimal set of points, good algorithms need to spend much effort searching for the "correct" circle – the circle containing the optimal set of points.

Figure 7: (Left) A histogram of the 50 best solutions found by `EnumAllCircles` showing how many solutions had the corresponding number of checklists previously reported to *eBird* (June, $k = 3$, $r = 20$ km). (Right) Trace map of the solutions for each month from January to June ($k = 5$, $r = 20$ km).

In order to show that the our techniques can also help in exploring under-sampled areas, we perform the following evaluation: We count the number of checklists submitted for regions within 5 km radius around the recommended locations from the 50 best solutions found by `EnumAllCircles`. Figure 7(Left) shows a histogram of the amount of existing checklists for these 50 best solutions, when considering the month of June, $k = 3$ and $r = 20$ km. The results reveal that while some solutions may already have a large number of checklists submitted (with the maximum near 150), most solutions contain locations in areas with few checklists (most solutions have less than 50 existing checklists, and many in fact have zero checklists). While some of these areas might be inconvenient to access, we argue that a lot of under-sampling results from a lack of attention, rather than inaccessibility. For example, Tompkins County, where the Cornell Lab of Ornithology is located, receives numerous checklists every year, while nearby Tioga county receives many fewer checklists, though it has a similar degree of accessibility. We hope our tool can direct bird watchers to under-sampled areas; thus improving the spatial coverage of *eBird* data and hence the quality of species distribution model.

Algorithm	# circles	runtime (secs)
`EnumAllCircles`	30,786,130	68,839
`EnumHexagonCircles`	333	2
`SampleCirclesGreedy`	10,000	15

Table 1: Runtime comparison of the different algorithms for `(k,r)-BestPlaces` (June, $k = 20$ and $r = 20$ km). The corresponding solution quality is shown in Figure 6(Left).

Finally, Figure 7(Right) shows the best regions found by `EnumAllCircles` from January to June ($k = 5$, $r = 20$ km), grouped by month. The places are clustered in the map, so we use one marker to represent all places recommended in each month. It is clearly seen that the best locations are moving farther north as the weather gets warm, which matches the known species migration patterns.

5 Conclusions

In this paper we address the task of identifying rewarding tasks to citizen scientists, while promoting scientific discovery. We developed *Bird-Watcher Assistant* to recommend interesting birding sites, aiming at boosting participation and scientific exploration of *eBird*. We propose two variants of the *probabilistic maximum coverage* problem, provide theoretical analysis of the two variants, describe several algorithms with provable approximation guarantees, as well as heuristic approaches, and provide empirical results using *eBird* data. Our algorithms are very fast and provide high quality solutions. Future directions include other variants of the problem with other types of constraints as well as the development of models that factor in the expertise level of the citizen scientists. We are also implementing our model as a mobile application.

6 Acknowledgments

This work was funded by the Leon Levy Foundation, the Wolf Creek Foundation, and the National Science Foundation (Grant Numbers OCI-0830944, CCF-0832782, ITR-0427914, DBI-1049363, DBI-0542868, DUE-0734857, IIS-0748626, IIS-0844546, IIS-0612031, IIS-1050422, IIS-0905385, IIS-0746500, IIS-1209589, AGS-0835821, CNS-0751152, CNS-0855167, CNS-1059284, DEB-110008).

A Appendix

Enumerating circles

We first introduce some notation. For any circle C, we call the set formed by all the points in P contained in C the point set *induced* by C, denoted as Q_C. The family of point sets $\mathcal{Q}_r$ is formed by all point sets that are induced by circles of radius r. More formally:

$$\mathcal{Q}_r = \{Q \subseteq P \mid \exists \text{ circle } C \text{ of radius } r, C \text{ induces } Q\}.$$

Any set S, that is a solution to the `(k,r)-BestPlaces` problem, satisfies the locality constraint and hence must be a subset of one member in $\mathcal{Q}_r$. On the other hand, any subset of a set in $\mathcal{Q}_r$ satisfies the locality constraint. Therefore, the `(k,r)-BestPlaces` problem could be rewritten as:

$$\begin{aligned}&\text{maximize } f(S)\\&\text{subject to } |S| \le k, \text{ and } \exists\, Q_C \in \mathcal{Q}_r, S \subseteq Q_C.\end{aligned}$$

If we have a way to enumerate all members in $\mathcal{Q}_r$, then for each such point set, the problem reduces to a submodular optimization subject to solely a cardinality constraint, for which we have an efficient approximation algorithm. Thus, the hardness of the problem is in how to enumerate all members in $\mathcal{Q}_r$. We define $\mathcal{Q}'_r$ as:

$$\mathcal{Q}'_r = \{Q \subseteq P \mid \exists \text{ circle } C \text{ of radius } r, C \text{ induces } Q, \text{ and}$$

C intersects with at least 2 points in $P\}$.

It is clear that $\mathcal{Q}'_r \subseteq \mathcal{Q}_r$. As we will prove in proposition A.1, $\mathcal{Q}_r \subseteq \mathcal{Q}'_r$ holds as well. Therefore, $\mathcal{Q}'_r = \mathcal{Q}_r$. This is encouraging because we have an obvious way to enumerate all members in $\mathcal{Q}'_r$ (or equivalently in $\mathcal{Q}_r$); namely, enumerating all pairs of points in P, and for each pair of distance at most $2r$ we form circles (possibly two) intersecting with this pair, and consider the point sets induced by these circles.

Proposition A.1. $\mathcal{Q}_r \subseteq \mathcal{Q}'_r$.

Proof. For any $Q \in \mathcal{Q}_r$, by definition, there exists a circle C_0 of radius r, such that C_0 induces Q. In the worst case, suppose all vertices in Q lie in the interior region of C_0. We can locally perturb C_0 until one vertex $q' \in Q$ hits the boundary of C_0. Note at this stage, C_0 still has one extra degree of freedom; therefore, we could continue moving C_0, while keeping q' fixed on its border, until another vertex $q'' \in Q$ falls into the boundary. The resulting circle is C_1. C_1 has at least q', q'' lying on its boundary, and contains the same set of vertices as C_0. In other words, C_1 induces Q as well. Thus, $Q \in \mathcal{Q}'_r$. This implies $\mathcal{Q}_r \subseteq \mathcal{Q}'_r$. □

Based on this proposition, we propose a polynomial time approximation algorithm, `EnumAllCircles`. This algorithm enumerates all members in $\mathcal{Q}'_r$. For each $Q \in \mathcal{Q}'_r$, the algorithm calls Algorithm 1 as an approximation procedure to get $S \subseteq Q$ with cardinality less than or equal to k. The set S with the best objective value across all members of $\mathcal{Q}'_r$ is returned.

Data: Point set $P = \{1, \ldots, n\}$, f, k and r.
Result: Point set S, $S \subseteq P$, $|S| \leq k$ and S is covered by a circle of radius r.

1 $S_{best} \leftarrow \emptyset$;
2 **for** *i, $j \in P$, and $dist(i, j) \leq 2r$* **do**
3   Find circle(s) C_1 (and potentially C_2) that intersects with both i and j;
4   **for** $C \in \{C_1, [C_2]\}$ **do**
5   extract points P_C covered by circle C;
6         $S \leftarrow$ Algorithm 1(P_C, f, k);
7   **if** $f(S) > f(S_{best})$ **then**
8             $S_{best} \leftarrow S$;
9   **end**
10   **end**
11 **end**
12 **return** S_{best}

Algorithm 3: `EnumAllCircles`: an `(1-1/e)` approximation algorithm for `(k,r)-BestPlaces`.

Algorithm 3 runs in $O(n^2(d + n_0 mk))$, where n is the number of points, n_0 is the maximal number of points within a circle, m is the number of species and d is the time to find the set of points that are induced by a circle. To minimize the time d, we use a KD-tree which stores all points according to their coordinates. Thus finding the set of points within a circle takes $d = O(\sqrt{n} + n_0)$.

`EnumAllCircles` achieves approximation bound of $(1-1/e)$ for the `(k,r)-BestPlaces` problem. Suppose the optimal value is OPT obtained on the point set S_{opt}. From the second formulation of the problem, there must exist $Q \in \mathcal{Q}_r$, such that $S_{opt} \subseteq Q$. Because $\mathcal{Q}_r = \mathcal{Q}'_r$, Q must be enumerated during the execution of Algorithm 3, i.e. there is a circle C considered by the algorithm such that P_C in line 5 is Q. Because S_{opt} is also the optimal solution to the `k-BestPlaces` problem when considering only points in Q, the solution S returned by the `k-BestPlaces` approximation algorithm 1 in line 6 during that iteration must satisfy $f(S) \geq (1-1/e)f(S_{opt}) = (1-1/e)OPT$. This implies that $f(S_{best}) \geq f(S) \geq (1-1/e)OPT$.

Enumerating Hexagons: Proof of Theorem 3.2

Proof. It is easy to see that S_{best} is covered by a circle of radius $2r$. To prove the approximation guarantee, assume R is a shape. It could be either a circle or a hexagon. Without causing confusion, we also use R to represent all the points contained in R. Denote by O_R an optimal set of points for `k-BestPlaces(R)`. Moreover, denote A_R as the set of points returned by Algorithm 1 when running on the set R. Hence, $f(A_R) \geq (1-1/e)f(O_R)$ for any shape R. Let $C_{tess}(r)$ be the set of circles circumscribing regular hexagons of side length $2r$ in the tessellation, as shown in Figure 3.

Given a `(k,r)-BestPlaces` instance, suppose C_{opt} is a circle of radius r that contains the optimal set of points S_{opt}. From Proposition 3.1, C_{opt} cannot intersect more than 3 hexagons. Without loss of generality, suppose C_{opt} intersects with hexagons H_1, H_2 and H_3. Let P_i be the points that are in $S_{opt} \cap H_i$. From the submodularity of $f(S)$, it follows that $OPT = f(S_{opt}) = f(P_1 \cup P_2 \cup P_3) \leq f(P_1) + f(P_2) + f(P_3)$. Again without loss of generality, assume $f(P_1) = \max\{f(P_1), f(P_2), f(P_3)\}$, then we have $OPT \leq 3f(P_1)$.

Note P_1 is a subset of points of size at most k and $P_1 \subseteq H_1$. Therefore, $f(P_1) \leq f(O_{H_1})$. Because C_1 circumscribes H_1, thus $f(O_{H_1}) \leq f(O_{C_1})$. Combining with $f(A_{C_1}) \geq (1-1/e)f(O_{C_1})$, we have $f(P_1) \leq (1 - 1/e)^{-1}f(A_{C_1})$.

Therefore, $f(A_{C_1}) \geq (1-1/e)f(P_1) \geq \frac{1}{3}(1-1/e)OPT$.

Finally, because $f(S_{best})$ is obtained by taking the maximum value $f(A_{C_j})$ for all circles C_j enumerated by the algorithm, we get the approximation bound.

□

Proof to Theorem 2.1

Proof. (Monotone) For any finite set B and element a

$$f(B \cup \{a\}) = NSpecies - \sum_{i \in Species} \prod_{j \in B \cup \{a\}} (1 - p_{i,j})$$

$$\geq NSpecies - \sum_{i \in Species} \prod_{j \in B} (1 - p_{i,j}) = f(B).$$

(Submodularity) For any finite set A, B, $A \subseteq B$, and an item a, $a \notin B$, it suffices to prove

$$f(A \cup \{a\}) - f(A) \geq f(B \cup \{a\}) - f(B). \quad (1)$$

It follows from the following calculation,

$$\begin{aligned}
& f(B \cup \{a\}) - f(B) \\
=& \sum_{i \in Species} (1 - \prod_{j \in B \cup \{a\}} (1 - p_{i,j})) - \\
& \sum_{i \in Species} (1 - \prod_{j \in B} (1 - p_{i,j})) \\
\leq & \sum_{i \in Species} \prod_{j \in A} (1 - p_{i,j}) p_{i,a} \\
=& \sum_{i \in Species} (1 - \prod_{j \in A \cup \{a\}} (1 - p_{i,j})) - \\
& \sum_{i \in Species} (1 - \prod_{j \in A} (1 - p_{i,j})) \\
=& f(A \cup \{a\}) - f(A).
\end{aligned}$$

□

Proof to Theorem 2.2

Proof. The decision version of `k-BestPlaces` is whether there is a set S, $S \subseteq P$, $|S| \leq k$, such that $f(S) \geq d$. We give a reduction from the Set Cover Problem. The Set Cover Problem is defined as: given an element set $R = \{r_1, r_2, \ldots, r_n\}$ and m subsets $R_1, R_2, \ldots, R_m$, are there k subsets $R_{i_1}, R_{i_2}, \ldots, R_{i_k}$ such that all elements in R are covered by $\cup_{j=1}^{k} R_{i_j}$?

Given a Set Cover instance, we construct a `k-BestPlaces` instance, where a species i corresponds to each element $r_i \in R$, and a location j corresponds to each set R_j. Consider the following deterministic setting, where $p_{ij} = 1$ if and only if $r_i \in R_j$; otherwise $p_{ij} = 0$. Finally, we set $d = n = |R|$.

Let S be the set of k locations corresponding to $R_{i_1}, R_{i_2}, \ldots, R_{i_k}$. It is sufficient to prove that $R_{i_1}, R_{i_2}, \ldots, R_{i_k}$ covers R if and only if $f(S) \geq n$. Given a `k-BestPlaces` solution S such that $f(S) \geq n$, for the sake of contradiction suppose that r_i is not covered by $R_{i_1}, R_{i_2}, \ldots, R_{i_k} \in S$. By our construction, this implies that $p_{ij} = 0$ for all location $j \in S$. Thus $1 - \prod_{j \in S}(1 - p_{ij}) = 0$, which implies $f(S) < n$. Conversely, given a Set Cover solution $R_{i_1}, R_{i_2}, \ldots, R_{i_k}$ that covers R, for the sake of contradiction suppose that the corresponding `k-BestPlaces` solution S has $f(S) < n$. Then, there must exist a species i, such that $p_{ij} = 0$ holds for all $j \in S$. By construction, this implies $r_i \notin R_j, \forall j \in \{i_1, \ldots, i_k\}$, i.e. r_i is not covered.

Finally, it is trivial to show `k-BestPlaces` admits a polynomial certificate, which completes our proof of NP-completeness.

□

References

Bartlett, T. 2011. Bugguide. bugguide.net.

Bonney, R.; Cooper, C. B.; Dickinson, J.; Kelling, S.; Phillips, T.; Rosenberg, K. V.; and Shirk, J. 2009. Citizen science: a developing tool for expanding science knowledge and scientific literacy. *BioScience* 59(11):977–984.

Călinescu, G.; Chekuri, C.; Pál, M.; and Vondrák, J. 2011. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.* 40(6):1740–1766.

Chekuri, C., and Pál, M. 2005. A recursive greedy algorithm for walks in directed graphs. In *FOCS*, 245–253.

Diggle, P. J. 2003. *Statistical Analysis of Spatial Point Patterns.* Hodder Education Publishers. 2nd edition.

Fink, D.; Hochachka, W. M.; Zuckerberg, B.; Winkler, D. W.; Shaby, B.; and et. al. 2010. Spatiotemporal exploratory models for broad-scale survey data. *Ecological Applications* 20(8):2131–2147.

Fink, D.; Damoulas, T.; and Dave, J. 2013. Adaptive spatio-temporal exploratory models: Hemisphere-wide species distributions from massively crowdsourced ebird data. In *AAAI*.

Homer, C.; Dewitz, J.; Fry, J.; Coan, M.; Hossain, N.; and et. al. 2007. Completion of the 2001 National Land Cover Database for the conterminous United States. *Photogrammetric Engineering and Remote Sensing* 73(4):337–341.

Kelling, S.; Gerbracht, J.; Fink, D.; Lagoze, C.; Wong, W.-K.; and et. al. 2012. ebird: A human/computer learning network for biodiversity conservation and research. In *IAAI*. AAAI.

Law, E., and von Ahn, L. 2011. *Human Computation*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

Lintott, C. J.; Schawinski, K.; Slosar, A.; Land, K.; Bamford, S.; and et. al. 2008. Galaxy Zoo: morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey. *Monthly Notices of the Royal Astronomical Society* 389(3):1179–1189.

Marshall, N. J.; Kleine, D. A.; and Dean, A. J. 2012. Coralwatch: education, monitoring, and sustainability through citizen science. *Frontiers in Ecology and the Environment* 10:332–334.

Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L. 1978. An analysis of approximations for maximizing submodular set functions I. *Mathematical Programming* 14(1):265–294.

Schwamb, M. E.; Orosz, J. A.; Carter, J. A.; Welsh, W. F.; Fischer, D. A.; and et. al. 2012. Planet hunters: A transiting circumbinary planet in a quadruple star system. *arXiv preprint arXiv:1210.3612*.

Singh, A.; Krause, A.; Guestrin, C.; and Kaiser, W. J. 2009. Efficient informative sensing using multiple robots. *J. Artif. Intell. Res. (JAIR)* 34:707–755.

Sullivan, B. L.; Wood, C. L.; Iliff, M. J.; Bonney, R. E.; Fink, D.; and Kelling, S. 2009. ebird: A citizen-based bird observation network in the biological sciences. *Biological Conservation* 142(10):2282 – 2292.

Weld, D. S.; Mausam; and Dai, P. 2011. Human intelligence needs artificial intelligence. In *Human Computation*, AAAI Workshops. AAAI.

Wood, C.; Sullivan, B.; Iliff, M.; Fink, D.; and Kelling, S. 2011. ebird: engaging birders in science and conservation. *PLoS biology* 9(12):e1001220.

Zhang, H.; Law, E.; Miller, R.; Gajos, K.; Parkes, D.; and Horvitz, E. 2012. Human computation tasks with global constraints. In *CHI*, 217–226.