Correction

COMPUTER SCIENCES

Correction for “Game-powered machine learning,” by Luke Barrington, Douglas Turnbull, and Gert Lanckriet, which appeared in issue 17, April 24, 2012, of *Proc Natl Acad Sci USA* (109:6411–6416; first published March 28, 2012; 10.1073/pnas.1014748109).

The authors note that, due to a printer’s error, the affiliation for Luke Barrington and Gert Lanckriet should instead appear as “Electrical and Computer Engineering Department, University of California at San Diego, La Jolla, CA 92093.” The corrected author and affiliation lines appear below. The online version has been corrected.

Luke Barrington[a], Douglas Turnbull[b], and Gert Lanckriet[a]

[a]Electrical and Computer Engineering Department, University of California at San Diego, La Jolla, CA 92093; and [b]Computer Science Department, Ithaca College, Ithaca, NY 14850

www.pnas.org/cgi/doi/10.1073/pnas.1205806109

Game-powered machine learning

Luke Barrington[a,1], **Douglas Turnbull**[b], **and Gert Lanckriet**[a]

[a]Electrical and Computer Engineering Department, University of California, La Jolla, CA 92093; and [b]Computer Science Department, Ithaca College, Ithaca, NY 14850

Edited* by Grace Wahba, University of Wisconsin-Madison, Madison, WI, and approved December 27, 2011 (received for review October 6, 2010)

Searching for relevant content in a massive amount of multimedia information is facilitated by accurately annotating each image, video, or song with a large number of relevant semantic keywords, or *tags*. We introduce game-powered machine learning, an integrated approach to annotating multimedia content that combines the effectiveness of *human computation*, through online games, with the scalability of *machine learning*. We investigate this framework for labeling music. First, a socially-oriented music annotation game called *Herd It* collects reliable music annotations based on the "wisdom of the crowds." Second, these annotated examples are used to train a supervised machine learning system. Third, the machine learning system *actively* directs the annotation games to collect new data that will most benefit future model iterations. Once trained, the system can automatically annotate a corpus of music much larger than what could be labeled using human computation alone. Automatically annotated songs can be retrieved based on their semantic relevance to text-based queries (e.g., "funky jazz with saxophone," "spooky electronica," etc.). Based on the results presented in this paper, we find that actively coupling annotation games with machine learning provides a reliable and scalable approach to making searchable massive amounts of multimedia data.

The last decade has seen an explosion in the amount of multimedia content available online: over 7 billion images are uploaded to Facebook each month (1), YouTube users upload 24 h of video content per minute (2), and iTunes, the world's largest music retailer, offers a growing catalog of more than 20 million songs (3). Developing a *semantic multimedia search engine*—that enables simple discovery of relevant multimedia content as easily as Internet search engines [e.g., Google (4)] help us find relevant web pages—presents a challenge because the domain of the query (text) differs from the range of the search results (images, video, music).

To enable semantic search of nontextual content requires a mapping between multimedia data and a wide vocabulary of descriptive *tags*. Describing multimedia content with relevant semantics necessitates intervention from humans who can understand and interpret the images, video, or music. However, manual tagging by human experts is too costly and time-consuming to be applied to billions of data items. For example, Pandora, a popular Internet radio service, employs musicologists to annotate songs with a fixed vocabulary of about five hundred tags. Pandora then creates personalized music playlists by finding songs that share a large number of tags with a user-specified seed song. After 10 y of effort by up to 50 full time musicologists, less than 1 million songs have been manually annotated (5), representing less than 5% of the current iTunes catalog.

Crowdsourcing has emerged as an affordable and scalable alternative to expert annotation by engaging many nonexpert contributors to label content online. Participants are motivated through small monetary rewards (6), or, even better, to contribute for free by disguising tasks as fun games, appealing to scientific altruism, or requiring it to access a service of interest. This distributed *human computation* has been applied; e.g., to categorize galaxies (7), fold proteins (8), transcribe old books (9), classify smiles (10) and apply descriptive tags to images (11), web pages (12) and music (13) (see *SI Text* for a review). Despite the promise of recruiting vast amounts of free labor, human computation *games* have had limited success in tagging the vast amount of multimedia content on the web: in 5 y, the ESPgame (11) has collected labels for up to 100 million images—roughly the same number that are uploaded to Facebook every 10 h—and TagATune (13) has labeled 30,000 song clips, or about 0.15% of iTunes' catalog.

Instead of requiring that humans manually label every image, video, or song, tagging can be partially automated using *supervised machine learning* algorithms that learn how semantics relate to multimedia. Machine learning approaches discover consistent patterns among a modest number of prelabeled training examples and then generalize this learned knowledge to label new, unlabeled data. The scalability of computer automation offers the potential to categorize massive amounts of multimedia information but reliability hinges on the *quality of training data* used. For example, by learning from millions of example images of faces in all possible poses, angles, and lighting conditions, machine learning algorithms (14) rapidly and reliably detect faces to automate focus in consumer digital cameras.

This paper proposes and investigates *game-powered machine learning* as a reliable and viable solution to annotating large amounts of multimedia content for semantic search, by leveraging the effectiveness of human computation through online games with the scalability of supervised machine learning. The main idea, illustrated for music search in Fig. 1, is to use an online *annotation game* to collect reliable, human-labeled examples that are tailored for training a supervised machine learning system. Once trained, this system can automatically annotate new content with the same tags used in the game, rapidly propagating semantic knowledge to lots of multimedia content. Through an *active learning* feedback loop, the game focuses on collecting data that most effectively improves future machine learning updates.

To validate the effectiveness of game-powered machine learning for *music* search, we designed and developed "Herd It," an online music annotation game that motivates players to contribute tags for songs. In contrast to previous "games with a purpose" which have aimed to annotate every image (11) or song (13, 15) on the web, Herd It was designed with a different, unique, and more realistic goal in mind: to enable the active machine learning approach presented in Fig. 1. To this end, Herd It was designed to integrate with a machine learning system that actively suggests songs and tags to be presented to users. As a result, the game can collect the most effective data for training the machine learning algorithm that then automates large-scale music tagging. Besides focusing human effort on efficient data collection, active song and tag suggestion also made gameplay more appealing.

Author contributions: L.B., D.T., and G.L. designed research; L.B. and D.T. performed research; L.B. contributed new reagents/analytic tools; L.B. analyzed data; and L.B., D.T., and G.L. wrote the paper.

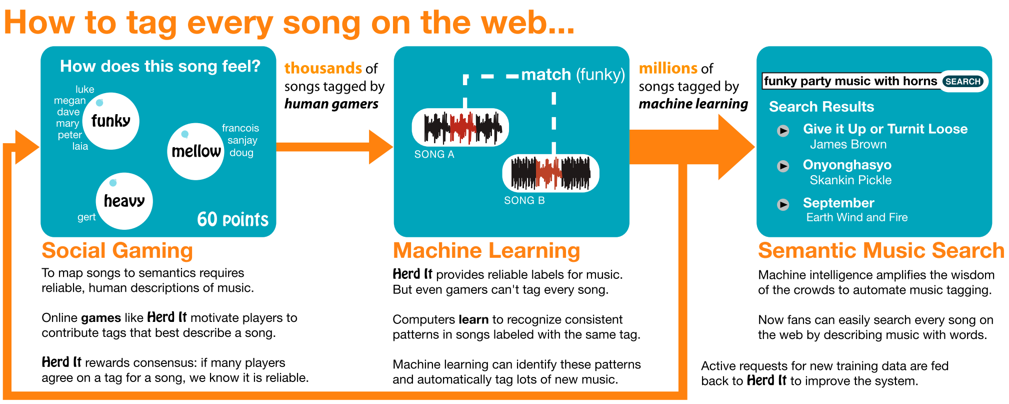
The authors declare no conflict of interest.

*This Direct Submission article had a prearranged editor.

Data deposition: Text dataset listing collected associations between popular music songs and descriptive text tags. Electronic datafiles containing signal processing features extracted from musical audio files.

[1]To whom correspondence should be addressed. E-mail: lukeinusa@gmail.com.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1014748109/-/DCSupplemental.



To map songs to semantics requires reliable, human descriptions of music.

Online **games** like **Herd It** motivate players to contribute tags that best describe a song.

Herd It rewards consensus: if many players agree on a tag for a song, we know it is reliable.

Herd It provides reliable labels for music. But even gamers can't tag every song.

Computers **learn** to recognize consistent patterns in songs labeled with the same tag.

Machine learning can identify these patterns and automatically tag lots of new music.

Machine intelligence amplifies the wisdom of the crowds to automate music tagging.

Now fans can easily search every song on the web by describing music with words.

Active requests for new training data are fed back to **Herd It** to improve the system.

Fig. 1. Game-powered machine learning framework for music annotation.

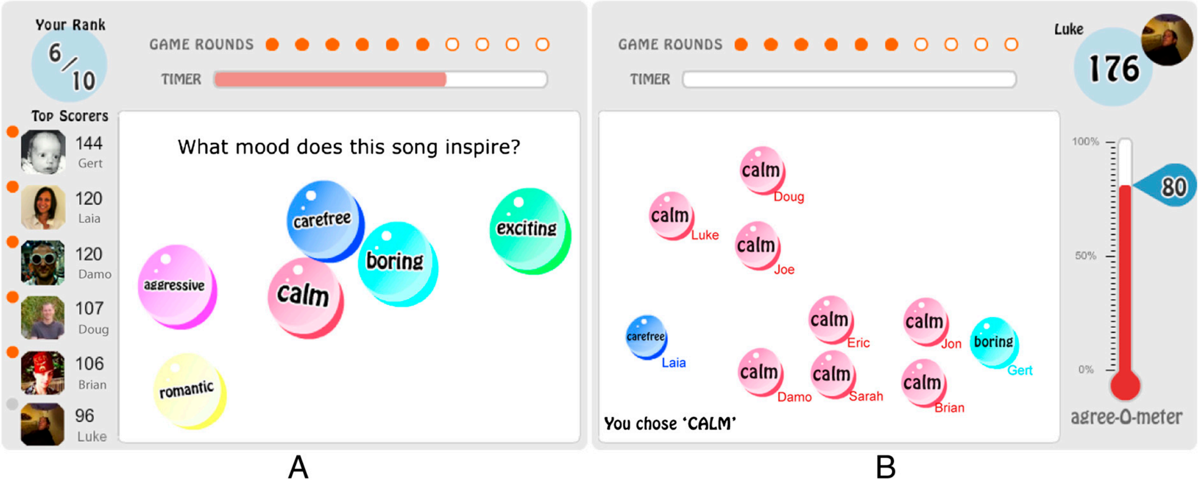
We deploy this game-based machine learning system to investigate and answer two important questions. *First*, we demonstrate that the collective wisdom of Herd It's crowd of nonexperts can *train* machine learning algorithms as well as expert annotations by paid musicologists. In addition, our approach offers distinct advantages over training based on static expert annotations: it is cost-effective, scalable, and has the flexibility to model demographic and temporal changes in the semantics of music. *Second*, we show that integrating Herd It in an active learning loop trains accurate tag models more effectively; i.e., with less human effort, compared to a passive approach.

Herd It—A Social Music Annotation Game

A player arriving at Herd It (www.HerdIt.org) is connected with "the Herd"—all other players currently online—and the game begins. Each round of Herd It begins by playing the same piece of music to all members of the Herd. A variety of fun, simple minigames prompt players to choose from suggested tags that describe different aspects of the music they hear (Fig. 2 illustrates an example of Herd It's gameplay with further examples in *SI Text*). In every minigame, players earn points based on their agreement with the tags chosen by the rest of the Herd, encouraging players to contribute tags that are likely to achieve consensus.

Herd It's goal is to collect training data that primes and improves the machine learning system through an active learning loop by motivating human players to provide reliable descriptions of a large number of example songs using a dynamic vocabulary of tags. To achieve this goal, Herd It's development followed a *user-centered design* process (16) that aimed to create an intuitive, viral game experience. A series of rapid prototypes were released every month and tested on focus groups of 5–50 new players, both in person at our lab and in a controlled online environment. During each test, we evaluated factors including playability and appeal, user-interface intuitiveness, viral potential, and stability. Interviews and questionnaires were used to evaluate the extent to which players were able to focus on the music (ensuring reliable data collection), their awareness of the other players (Herd It is a social game and a player's score depends on the Herd), and overall enjoyment (indicating likelihood of large-scale participation). Iterative user feedback led to improvements in the design and the process continued until key gameplay and social evaluation metrics were satisfied (e.g., 94% of players understood the scoring metric within five games, 82% said they would recommend Herd It to their friends; further results in *SI Text*). The user-centered design process was instrumental in determining crucial gameplay mechanics, described below, that differentiate Herd It from other music annotation games [e.g., (13, 15)].

In particular our user tests discovered that, while free-text tagging works well when annotating images (which tend to feature many obvious, easily named objects; see; e.g., ref. 11), many listeners found it difficult to produce and agree on a variety of tags for music in a game environment without some priming. Asking players to type their own descriptions of the music meant that the vast majority of tags were confined to a limited vocabulary of generic tags; e.g., "rock," "guitar," "drums," "male/female vocalist" [MajorMiner (15) suffers from this problem]. As a result,


What mood does this song inspire?

You chose 'CALM'

Fig. 2. Illustration of Herd It gameplay. (*A*) Six bubbles float around the play area, each suggesting a mood that might be evoked by the music that is playing. The player clicks the bubble they feel is most appropriate and all other bubbles disappear with a pop. After 15 s, the minigame ends. (*B*) Choices made by the rest of the Herd are revealed. During this feedback period, an "agree-O-meter" fills up as other members of the Herd agree with the player's choice. Players earn points equal to the percentage of the Herd in agreement with them, rewarding consensus and implicitly collecting reliable music tags.

the independent inputs of multiple players rarely converged on more interesting tags [TagATune (13) avoids this problem by asking players to guess whether they are listening to the same song, based on the free-text tags other players entered, rather than requiring agreement on the exact tags]. To achieve both variety and consensus, Herd It's unique solution is to *suggest* tags for player confirmation, thereby controlling the vocabulary used to describe music while maintaining simple and compelling gameplay. In addition, tag suggestion addresses another, important design objective: it facilitates the active learning paradigm depicted in Fig. 1 which requires precise control over the data collected from the Herd. Specifically, an active learning approach leverages machine learning models to suggest {song,tag} combinations that, if confirmed by human players, are most likely to produce useful training examples and optimize future model training. Herd It's tag suggestion mechanism enables active learning by focusing human labeling on specific {song,tag} combinations. Vice versa, Herd It's new tag suggestion design benefits from it being powered with machine intelligence. Indeed, suggesting tags randomly, rather than intelligently, was found to result in many minigames that have no relevant choices and are not fun.

To achieve widespread player engagement and thus maximize training data collection, we found that Herd It should target the "casual" gamer. Unlike the traditional computer gaming demographic (i.e., teenage boys) who enjoy long-lasting games with complicated gameplay mechanics, casual games appeal to a much wider demographic (e.g., skewed towards middle-aged women), are played in short time increments (5–20 min) and feature simple but addicting gameplay (17). Herd It's simple, single-click gameplay, cartoon-ish minigame design, and intuitive scoring metric were designed to attract a broad audience of casual gamers.

Based on the choices offered in a given minigame, different users may end up describing a song differently, using either compatible tags (e.g., a "romantic" song that is also described as "carefree") or opposite tags (e.g., what sounds "exciting" to one listener may be "boring" to another). Given this subjectivity inherent in music appreciation, our design process revealed that it is important to evaluate agreement in minigames in a (larger) group setting, as this enables clusters of consensus to develop between the players, around multiple "right" answers. This observation inspired us to make "the Herd" a central feature of the game, rather than the player-vs-player mechanic used by other games [e.g., (11, 13)]. In addition, our user tests determined that *realtime*, social interaction produced more compelling gameplay than off-line group feedback [e.g., (15)]. The group dynamic also makes it more difficult for a few players to cheat and gain lots of points by coordinating poor labeling (other measures to prevent cheating include randomizing tag order in minigames and preventing a single player from entering multiple games).

Finally, because individuals use music preference to communicate information about their personality (18), players desired Herd It to be embedded in a larger social music experience. For example, players wanted to choose preferred genres, share music, create personal profiles, and challenge and compare scores with friends, leading us to integrate the game within the players' existing social network by releasing Herd It as an application on Facebook. Integrating Herd It with Facebook offers many avenues to engage players (e.g., easy login, personalized messages, player photos) and promote the game to a wide audience (e.g., invites, challenges, see *SI Text*). Facebook also provides demographic and psychographic information about players (e.g., gender, age, location, friends, favorite music), offering a hitherto unavailable level of insight into how different people experience and describe music.

Automatic Music Tagging

Statistical pattern recognition methods for tagging music begin by extracting *features* that summarize properties of the acoustic waveforms, essentially "listening" to the musical signal. By considering a training set of reliably labeled songs, supervised machine learning algorithms identify statistical regularities in these acoustic features that are predictive of descriptive tags like "bluegrass," "banjo," or "mellow." Machines can then generalize this knowledge by detecting the presence of similar patterns in vast catalogs of new, untagged music, thereby leveraging the accuracy of human labeling (to obtain the training set) with the scalability of automated analysis to tag this new music content. Machine learning methods for music tagging continue to improve and, given training data of sufficient quality, their accuracy approaches the ceiling set by the inherent subjectivity in describing music with tags (19).

To thoroughly evaluate the efficacy of the game-powered machine learning paradigm depicted in Fig. 1, we consider various state-of-the-art autotagging algorithms for its machine learning component, including generative (19, 20) and discriminative (21, 22) approaches. Generative methods focus on estimating the (class-conditional) distribution [e.g., with a Gaussian mixture model (GMM), dynamic texture mixtures (DTM), etc.] of acoustic features that are common among songs that human "trainers" have labeled with a given tag. By evaluating the likelihood of features from a new song under the learned distribution, the model determines the probability that the tag is a relevant description of the song (19). Discriminative methods, on the other hand, directly optimize a decision rule to discriminate between a tag being present or absent for a given audio clip. Evaluating the decision rule for a new song allows to obtain tag probabilities. Just as Internet search engines rank web pages by their relevance to a text query, the tag probabilities output by a model can be used to rank songs by their relevance to the tag.

Traditional machine learning approaches use a single, fixed training set to learn models that, once trained, remain static. In our game-powered machine learning framework however, new data is constantly being contributed by Herd It players. That data can be used to update our tag models. Even more, because Herd It's design permits actively focusing players' efforts on specific songs and tags, it is possible to collect specifically that data that is expected to improve tag models most effectively, achieved through an *active learning* approach (see ref. 23 for a review), that leverages the current tag models to identify the most effective song-tag pairs for future model updates. Active learning fully integrates the autotagging algorithm with the data collection process, to optimize model training. To investigate the benefits of deploying Herd It in an active learning loop compared to updating with randomly collected data, we develop a unique active learning algorithm to suggest data for training the generative GMM-based autotagger, a top performer in the 2008 MIREX evaluation of automatic music tagging algorithms (24).

Various active learning algorithms have been proposed for *discriminative* machine learning methods[†], where both positive and negative examples are used to learn a decision boundary between classes. *Generative* approaches, on the other hand, require only positively labeled examples for training (i.e., songs that exemplify a certain tag) and negatively labeled training examples offer no improvement to the model[‡]. To collect positively labeled training examples and improve a generative model through active learning may suggest sampling unlabeled examples that have high likelihood under the current model and procuring labels for them [e.g., by presenting {song,tag} pairs in Herd It minigames]. However, this "certainty" sampling approach suffers from two drawbacks: early in training, when the model is not yet well learned,

[†] Strategies for actively learning discriminative models include *uncertainty sampling* (25) where points are chosen that are least certain (or have highest entropy) under the current model (e.g., points closest to the decision boundary) and *variance reduction* (26) where samples are chosen to reduce the model's output variance.

[‡] For example, for generative models, uncertainty sampling faces the problem that unlabeled songs which have low certainty under the current model are likely to result in negative labels.

the most likely samples may not in fact be positive examples and thus will not contribute to the training set. Later in the learning process, sampling from the most likely areas results in many confirmed positive examples that conform to the model's current training set and lack the diversity required to generalize the current model to uncertain areas of the feature space. Exploration of these uncertain areas advocates for a more random sampling of unlabeled examples. Rather than a complete random sampling, we can actively increase the efficiency of the data collection and, thus, the learning rate, by reducing the likelihood of sampling unlabeled examples that are eventually labeled as negatives (which are of no use to train the generative model) and thereby avoiding points that most disagree with the current model. More specifically, we rank all of the unlabeled examples by their likelihood under the current model, remove the 10% of examples with lowest likelihood, and query labels randomly from the remaining 90% of examples. By removing the least likely points and sampling randomly elsewhere, we aim to avoid querying labels for negative examples and achieve rapid confirmation of a diverse training set for our generative model. Our active GMM experiments show that removing the 10% least likely songs finds a good balance between exploring poorly modeled areas of the feature space while avoiding points that are unlikely to produce positive examples.[§]

Game-Powered Machine Learning

Our game-powered machine learning approach aims to collect sufficient human labels, through game play, to train an automatic music annotation system that can reliably generalize semantics to unlimited music. Qualitatively, we argue that this crowdsourced approach is superior to requiring expert annotators as it is less costly, more scalable, and collects a dynamic dataset that can be adapted over time to focus on the most relevant or important tags. To quantify the efficacy of our game-powered machine learning framework, we conduct experiments designed to answer the following two questions: (*i*) Can machine learning algorithms be trained with data collected from Herd It's crowd of nonexperts as accurately as with data collected from paid expert musicologists? (*ii*) Can accurate tag models be learned with less human effort by encapsulating Herd It in an active learning framework? To answer these questions, we deployed Herd It online, engaging 7,947 people to provide over 140,000 clicks that associate songs with tags through five different types of minigames.

To generate minigames in a passive system, without active learning, we begin with 10–20 candidate {song,tag} pairs, chosen randomly from the authors' personal music collection of over 6,000 popular songs from the past 70 y, and a vocabulary of 1,269 tags, including subgenres, emotions, instruments, usages, colors, and more categories. To generate a minigame, one {song,tag} pair is selected from the list of candidate pairs, biased by associations (determined using the online music service http://last.fm/api/) with the musical genre selected by the player at the start of the game (pop, rock, hip-hop, blues, electronica, or "everything") and by the particular minigame (e.g., certain minigames focus on subgenres, colors, or bipolar adjectives). Remaining minigame tags (each minigame suggests between one and nine tags) are restricted to the same tag category. Candidate {song,tag} pairs remain on the list until they have been viewed by at most 50 players. At that point the {song,tag} pair is discarded and replaced by a new, randomly sampled one. Maintaining a reasonable list of candidate pairs ensures diverse gameplay.

Consensus between players' clicks collected in Herd It minigames is used to "confirm" reliable {song,tag} associations. More specifically, the generative model of labels, accuracies, and difficulties, or "GLAD," (10) conceives of each human input as an estimate of the underlying true label that has been corrupted by player inaccuracy and the difficulty of labeling the song. Using an expectation-maximization algorithm, GLAD optimally combines the votes from all Herd It players and we confirm the findings of (10) that the resulting consensus is more reliable than heuristics such as majority vote, percentage agreement, or vote thresholds. A {song,tag} pair is presented in Herd It minigames until GLAD "confirms" a reliable association, based on the historical click data for that {song,tag} pair. If a {song,tag} pair remains unconfirmed after being viewed by 50 Herd It players, it is "rejected" and not sampled further. Overall, GLAD confirmed 8,784 {song,tag} pairs, representing song examples of 549 tags, while 256,000 pairs were rejected. To ensure that we have enough data to train robust machine learning models and answer the first question, we reduce the dataset to the 127 tags for which Herd It has identified at least 10 reliable example songs. Data was collected passively (i.e., no active learning) and this provides the baseline against which to compare an *active* learning strategy and evaluate the second question.

To answer the first question, we quantify the efficacy of our game-powered machine learning framework and compare it to "expert-trained" machine learning. That is, we evaluate the performance of a music autotagging algorithm when trained on (*i*) the Herd It *game* data and (*ii*) data derived from *expert* musicologists at Pandora.com's "Music Genome Project" (*MGP*), respectively. After training, the accuracy of each autotagger is evaluated on *CAL500*: an independent evaluation set of 500 songs fully labeled by multiple humans using a controlled survey (19) (see *SI Text* for details about the *CAL500* and *MGP* datasets). For this comparison, we train and evaluate models of all tags that are available in both the *MGP* and *CAL500* vocabulary and for which Herd It has collected at least ten confirmed example songs, resulting in 25 tags. The models of each of these tags are used to retrieve the 10 most relevant songs from the *CAL500* corpus for each single-tag query. These top-ten search results—automatically retrieved by a machine—are evaluated by comparing to the *CAL500 ground-truth*, and computing the precision (i.e., the number of songs in the machine-ranked top-ten that the ground truth effectively associates with the tag). Finally, the precision is averaged over all 25 tags. Because both Herd It and *MGP* models are instances of the *same* machine learning algorithm, but trained on *different* datasets, any significant differences in autotagging performance most likely reflect differences in the quality of the respective training data sources and allow us to evaluate human computation games—Herd It, in particular—as a source of reliable training data. To prevent bias induced by a particular choice of machine learning algorithm, this comparison is repeated for multiple state-of-the-art autotagging algorithms.

In addition to showing that game-powered machine learning can be competitive with an expert-trained system, in a second step, we demonstrate the efficacy of *actively* integrating machine learning with game-based data collection. The baseline here is the passive approach outlined above, which "analyzes" (i.e., confirms or rejects through human labeling) {song,tag} pairs in random order. As more {song,tag} pairs are analyzed (i.e., more human effort contributed), a tag's training set grows, tag models are updated and autotagging performance is expected to improve. We compare this passive approach to an active learning paradigm which aims to improve tag models more effectively by leveraging current models to select the next {song,tag} pairs that will be analyzed. More precisely, for each of the 25 Herd It tags that were evaluated earlier, we collect all songs that appeared with the tag (confirmed or rejected) in previous Herd It minigames. We then estimate 25 GMM-based tag models by engaging in an iterative training procedure, for each tag, based on this list of "candidate" songs. At each iteration, we first compute the likelihood, under the current tag model, of all remaining candidate songs and use our active learning method for

[§] In a feature space of high dimension, d, the probability density of a Gaussian distribution with variance σ is focused on a small shell a distance $\sigma\sqrt{d}$ from the mean. Thus the majority of points tend to have very similar GMM likelihoods (27). While this fact can make it difficult to identify positive points based on likelihood, any points that have significantly *lower* than average likelihood can be excluded with confidence.

generative models to prioritize 10 candidate songs for analysis with that tag (for the first iteration, candidate songs are chosen randomly). That is, we use our active learning algorithm to resample {song,tag} pairs that were previously presented in Herd It games. Songs for which the {song,tag} pair was previously confirmed are added to the tag's training set; the remaining, rejected songs are removed from future candidate lists. Finally, we retrain the tag model using the updated training set and evaluate its performance on the *CAL500* test set. We once again recompute the likelihood of all remaining candidate songs under the updated model, actively select 10 candidate songs for analysis, retrain the tag models, and so on. Song selection is repeated up to 200 times, analyzing up to 2,000 songs for each tag. At each iteration, we evaluate the average performance of the 25 tag models. We compare active learning to the passive baseline which samples 10 songs randomly at each iteration, for each tag.

Results

Table 1 presents the average precision of the top-ten music search results for 25 single-tag queries on *CAL500*, achieved by training four state-of-the-art autotagging algorithms on Herd It's data. We compare to the performance obtained by training on expert *MGP* data. For each of the 25 tags common to Herd It, *MGP* and *CAL500*, we evaluate the top-ten precision on *CAL500* and average performance over all tags. While the absolute performance depends on the machine learning method used, the *relative* performance between models trained using Herd It and those that use *MGP* data remains consistently over 95%. These findings answer our first question by demonstrating that a game-based machine learning system, trained on data collected from Herd It players, provides a competitive alternative to a system trained on expert labeled data, across a variety of algorithms.

Fig. 3 offers a more detailed comparison of Herd It and *MGP* based systems, by examining the performance of each tag model learned by the hierarchical GMM algorithm (19). The ability of the machine learning algorithm to model different tags varies; e.g., "acoustic," "male lead vocals," and "hip hop" songs are more easily identified, while "hand drums" and "funk" music are poorly modeled. In general, model performance is independent of the training data source (i.e., most points lie close to the diagonal in Fig. 3, indicating comparable results for each system). Models trained on either data source performed significantly differently for just two tags: "synthesizer" (*MGP*-based model better) and "drum set" (Herd It-based model better, 2-tailed t-test, 95% significance level). In summary, Table 1 and Fig. 3 quantitatively demonstrate that training from Herd It's crowdsourced data captures knowledge similar to training from expert annotations.

We turn now to the second question: can integrating machine learning and Herd It's game-powered data collection in an active learning loop train accurate models with less human effort than a passive system? To measure human effort, we consider the number of {song,tag} pairs analyzed through Herd It gameplay, expressed as the number of songs analyzed per tag (for each of the 25 tags being modeled). Fig. 4 displays the improvement in song retrieval performance of the GMM autotagging algorithm as more songs are analyzed for each tag (and, consequently, more training examples collected for model estimation), following both an active learning and a random sampling strategy. The results demonstrate an improved learning rate due to active learning: active learning requires analyzing, on average, 450 songs per tag to achieve no significant difference between Herd It and *MGP* performance (paired, one-tailed t-test, $p = 0.1$) while the passive strategy hits this level after analyzing 940 songs for each tag. Fig. 4 highlights the improved efficiency by shading the learning curves while performance is significantly different from the *MGP*: by prioritizing the order in which {song,tag} pairs are presented to players, our active learning approach reduces the human labeling effort required by half. A more detailed inspection of the results reveals that active learning achieves expert performance by confirming an average of 31 training songs per tag, out of 450 analyzed candidates, vs. 49 out of 940 for random sampling. Active learning boosts the learning rate by suggesting fewer {song, tag} pairs that are eventually rejected and not used for training [compared to suggesting random {song,tag} pairs] while still producing a sufficiently diverse set of confirmed training songs.

Table 1. Average top-ten precision of autotagging algorithms trained on Herd It examples and tested on *CAL500*

Autotagging algorithm	Top-10 Precision Herd It training	Relative Precision Herd It : *MGP*
Hierarch. GMM (19)	0.40	95.8% ± 4.6
Hierarch. DTM (20)	0.42	98.9% ± 6.0
Boosting (21)	0.38	99.7% ± 4.2
SVM (22)	0.38	95.6% ± 7.2

Also shown is relative performance (top-ten precision) of Herd It-trained models compared to models trained on expert *MGP* examples. Top-ten precision of random guessing is 0.18.

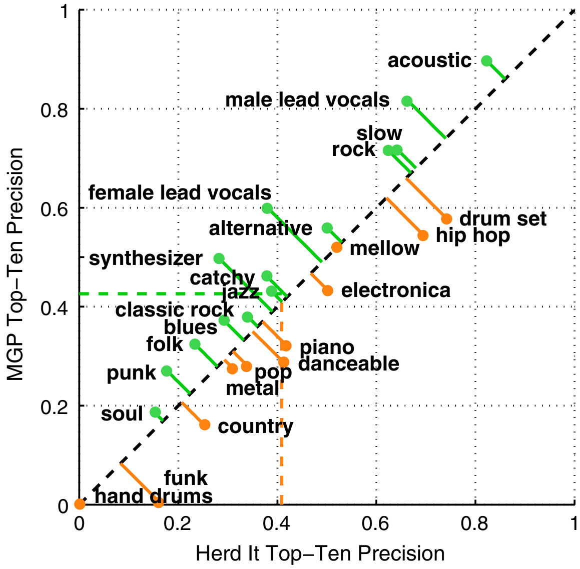


Fig. 3. Top-ten precision for machine learning models trained on Herd It's crowdsourced data (*x*-axis) and models trained on data from the Music Genome Project (*y*-axis). While absolute performance depends on the tag (e.g., "acoustic" music is better modeled than "soul" music), on average (dashed lines) Herd It's crowdsourced data trains models that are as precise, at the tag-level, as models learned from expert-labeled data.

Having demonstrated that Herd It data can train automatic music taggers that are as accurate as an expert-trained system, we now compare the tagging efficiency of crowdsourced amateur players with that of trained experts. Pandora's musicological experts take 20–30 min to analyze and quantify the association between a song and 100–500 semantic dimensions, a rate of about 12 song-tag associations per expert-minute. Herd It minigames last about 30 s and present, on average, 5.4 tags for player analysis. Thus a single Herd It player analyzes 10.8 song-tag associations per minute, a little less than the Pandora expert. To quantify (i.e., confirm or reject) a song-tag association, the analysis of up to 50 players is required, vs. that of one Pandora expert. So, Herd It's game-based approach gathers reliable tags from humans for free with about 2% the efficiency of paid, expert labeling. Of course, Herd It's lower efficiency is multiplied by the number of simultaneous players in the Herd, which could be significantly larger than the number of musicological experts that can be gainfully employed, simultaneously.

In comparing game-based and expert annotation methods, we recognize that, even with crowdsourced consensus, multiple

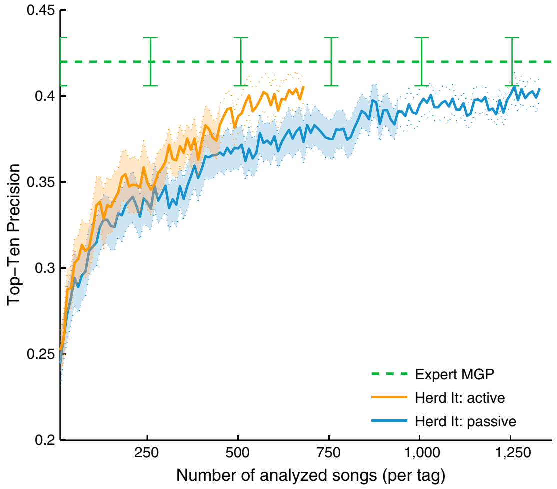


Fig. 4. Music autotagging performance as a function of human effort (i.e., number of songs analyzed by Herd It players). For each of the 25 tags considered, "passive" randomly selects songs for analysis while "active" leverages an active learning paradigm. Each {song,tag} pair appeared in Herd It minigames until it was either confirmed by GLAD or rejected after being presented to 50 players. The *y*-axis plots the average precision of the top-ten search results returned by tag models trained on all songs confirmed by Herd It at each 10-song increment on the *x*-axis until the target performance reported in Table 1 (for batch training) is achieved (680 songs for active, 1,330 songs for passive). Error bands show the standard error of the mean, averaged over three independent trials, and remain shaded while Herd It performance is significantly below *MGP* (paired t-test, $p = 0.1$). Integrating active learning with Herd It's data collection improves the learning rate, achieving significant performance with less human effort.

amateur raters are likely less reliable than experts when identifying certain details pertaining to musical theory (e.g., we find Herd It players are inconsistent in Scales minigames that ask to distinguish major from minor keys) or esoteric subgenres and instruments (e.g., tags for the subgenres "doom metal" and "worldbeat" and the instruments "siren" and "spoons" were rarely chosen when suggested in a minigame and, if they were chosen, it often was seemingly without relation to the audio content). In designing Herd It's games, we generally focused on tags that are more relevant to our goal of building a music search engine that can empower a wide audience to discover relevant music using simple, semantic search.

Finally, while a human-only approach requires the same labeling effort for the first song as for the millionth, our game-powered machine learning solution needs only a small, reliable training set before all future examples can be labeled automatically, improving efficiency and cost by orders of magnitude. Tagging a new song takes 4 s on a modern CPU: in just a week, eight parallel processors could tag 1 million songs or annotate Pandora's complete song collection, which required a decade of effort from dozens of trained musicologists.

Conclusions

We proposed game-powered machine learning as an integrated, scalable, affordable, and reliable solution for semantic search of massive amounts of multimedia content and investigated its efficacy for music search. Herd It, an online music annotation game, collects reliable examples of how humans use semantic tags to describe music. By itself, this human computation approach is insufficient to label the millions of songs available on the web. Instead, the knowledge collected by our game trains machine learning algorithms that can generalize tags to vast amounts of new, unlabeled music. Compared to other music games with a purpose, Herd It was specifically designed to be actively integrated with the machine learning algorithms and provide the data that most effectively trains them. Our results demonstrate, first, that game-powered machine learning is as good as expert-based machine learning—annotations collected from human computation games train autotagging models as accurately as expensive, expert annotations—while offering some distinct advantages (e.g., cost-effectiveness, scalability, flexibility to update the game to focus on tags of interest). Second, we show that embedding Herd It in an active learning paradigm trains accurate autotaggers more effectively; i.e., with less human effort, compared to a passive approach. We conclude that actively integrating human computation games and machine learning—combining targeted data collection by annotation games with automatic prediction by scalable machine learning algorithms—enables simple, widespread multimedia search and discovery.

ACKNOWLEDGMENTS. The authors thank Damien O'Malley for assistance in the design, development, and testing of Herd It, and Sanjoy Dasgupta and Brian McFee for active learning advice. L.B. and D.T. received funding from a National Science Foundation (NSF) fellowship (DGE-0333451). L.B. received funding from the Qualcomm Innovation Fellowship. L.B. and G.L. received funding from the Hellman Fellowship Program, the von Liebig Center, the Committee on Research (grant RJ138G-LANCKRIET), the Alfred P. Sloan Foundation, Yahoo! Inc., and the NSF (DMS-MSPA 0625409 and IIS-1054960).

1. Facebook (2012) *http://www.facebook.com/press/info.php?statistics* February.
2. YouTube (2012) *http://www.youtube.com/t/fact_sheet* February.
3. Apple iTunes (2012) *http://www.apple.com/itunes/features/* February.
4. Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. *Comput Netw* 30:107–117.
5. Pandora Media Inc. (2011) Securities and Exchange Commission Form S-1., http://www.sec.gov/Archives/edgar/data/1230276/000119312511032963/ds1.htm February.
6. Amazon Mechanical Turk (2012) *http://mturk.amazon.com*.
7. GalaxyZoo (2012) *http://www.galaxyzoo.org*.
8. Cooper S, et al. (2010) Predicting protein structures with a multiplayer online game. *Nature* 466:756–760.
9. von Ahn L, Maurer B, McMillen C, Abraham D, Blum M (2008) reCAPTCHA: human-based character recognition via web security measures. *Science* 321:1465–1468.
10. Whitehill J, Ruvolo P, Bergsma J, Wu T, Movellan J (2009) Whose vote should count more: optimal integration of labels from labelers of unknown expertise. *In 23rd Conference on Neural Information Processing Systems (NIPS)* (MIT Press, MA).
11. von Ahn L, Dabbish L (2004) Labeling images with a computer game. *In 22nd International Conference on Human Factors in Computing Systems* (ACM SIGCHI).
12. von Ahn L (2006) Games with a purpose. *IEEE Computer Magazine* 39:92–94.
13. Law E, von Ahn L (2009) Input-agreement: a new mechanism for collecting data using human computation games. *In 27th International Conference on Human Factors in Computing Systems* (ACM SIGCHI).
14. Viola P, Jones M (2004) Robust real-time face detection. *Int J Comput Vision* 57:137–154.
15. Mandel M, Ellis D (2008) A web-based game for collecting music metadata. *J New Music Res* 37:151–165.
16. Gould J, Lewis C (1985) Designing for usability: key principles and what designers think. *Commun ACM* 28:300–311.
17. Rohrl D, ed. (2008) Casual Games White Paper. (International Game Developers Association).
18. Rentfrow P, Gosling S (2006) Message in a ballad: the role of music preferences in interpersonal perception. *Psychol Sci* 17:236–242.
19. Turnbull D, Barrington L, Torres D, Lanckriet G (2008) Semantic annotation and retrieval of music and sound effects. *IEEE T Acoust Speech* 16:467–476.
20. Coviello E, Barrington L, Lanckriet GRG, Chan AB (2010) Automatic music tagging with time series models. *In 11th International Society for Music Information Retrieval (ISMIR) Conference* (International Society for Music Information Retrieval , Netherlands).
21. Eck D, Lamere P, Bertin-Mahieux T, Green S (2007) Automatic generation of social tags for music recommendation. *In 21st Conference on Neural Information Processing Systems (NIPS)* (MIT Press, MA).
22. Mandel M, Ellis D (2008) Multiple-instance learning for music information retrieval. *In 9th International Society for Music Information Retrieval (ISMIR) Conference* (International Society for Music Information Retrieval , PA).
23. Settles B (2010) Active learning literature survey. *University of Wisconsin—Madison: Computer Sciences Technical Report.*
24. Downie JS (2008) Audio tag classification. *Music Information Retrieval Evaluation eXchange (MIREX)*, http://music-ir.org/mirex/wiki/2008:Audio_Tag_Classification_Results.
25. Lewis D, Catlett J (1994) Heterogeneous uncertainty sampling for supervised learning. *In 11th International Conference on Machine Learning (ICML)* (Morgan Kaufmann, CA).
26. Cohn DA, Ghahramani Z, Jordan MI (1996) Active learning with statistical models. *Journal of Artificial Intelligence Results* 4:129–145.
27. Dasgupta S, Schulman LJ (2007) A probabilistic analysis of EM for mixtures of separated, spherical Gaussians. *J Mach Learn Res* 8:203–226.