2015 IEEE Symposium on Service-Oriented System Engineering

Towards an Effective Crowdsourcing Recommendation System

A Survey of the State-of-the-Art

Eman Aldhahri, Vivek Shandilya, Sajjan Shiva
Computer Science Department,
The University of Memphis
Memphis, USA
{aldhahri, vmshndly, sshiva}@memphis.edu

Abstract—Crowdsourcing is an approach where requesters can call for workers with different capabilities to process a task for monetary reward. With the vast amount of tasks posted every day, satisfying workers, requesters, and service providers--who are the stakeholders of any crowdsourcing system--is critical to its success. To achieve this, the system should address three objectives: (1) match the worker with a suitable task that fits the worker's interests and skills, and raise the worker's rewards; (2) give requesters more qualified solutions with lower cost and time; and (3) raise the accepted tasks rate which will raise the aggregated commissions accordingly. For these objectives, we present a critical study of the state-of-the-art in recommendation systems that are ubiquitous among crowdsourcing and other online systems to highlight the potential of the best approaches which could be applied in a crowdsourcing system, and highlight the shortcomings in the existing crowdsourcing recommendation systems that should be addressed .

Keywords—Crowdsourcing; recommendation; Survey; task matching.

I. INTRODUCTION

A crowdsourcing paradigm facilitates outsourcing tasks to a pool of workers distributed around the world [17]. However, matching workers with suitable tasks is a serious obstacle due to the tremendous multitude of tasks and workers available on crowdsourcing systems [6, 12].

Obviously, finding the suitable task between tens of thousands of tasks is a very time consuming process. Appropriateness of tasks depends mainly on two factors: interest and capability of the worker. Worker interest is measured based on multidimensional factors that are weighted differently by each worker, such as monetary reward and task type [14].

Various areas contribute to building recommendation systems: cognitive science, approximation theory, information retrieval, forecasting theories, management science, and marketing [15]. The first step in any recommendation system is building the user's preferable profile either explicitly or implicitly. Implicit profiles are based on users' previous behaviors. User behavior can be interpreted as preferable features.. Explicit profiles are based on asking users to complete a preferable features form. Two techniques common in this area are: a collaborative filtering approach (with cold start problems, scarcity and scalability in large datasets [6]), and a content-based approach (with problems due to overspecialization, etc.). A hybrid approach that combines the two techniques has also been used.

II. MOTIVATION

The main objective in this study is to investigate various online recommendation systems by analyzing their input parameters, effectiveness, and limitations in order to assess their usage in crowdsourcing systems. In other words, how can we derive the best practices from various recommendation systems, which share some common features with crowdsourcing systems, and harness these practices to model an effective recommendation system? We concentrate on seven factors to distinguish between each of the efforts to achieve their objectives:

1- What parameters of the system are used in formulating the problem?
2- What is the "computational problem" formulated to make recommendations?
3- How does the problem's formulation handle the private imperfect/ information of the stakeholders?
4- How is the problem solved?
5- How is the solution implemented to construct the recommendation system?
6- How scalable is the solution?
7- What are the limitations of the work?

III. RECOMMENDATON SYSTEMS

In this section, we present a critical review of the works as shown in Fig.1. We classify them based on their main contribution in the methodologies and technologies associated with the recommendation system. In the reviews, we present a general summary of each, identify the main contribution, and evaluate how they address the seven questions we formulated in section II.

978-1-4799-8356-8/15 $31.00 © 2015 IEEE
DOI 10.1109/SOSE.2015.53

A. General Recommendation Papers

In this section, we explore some of the online systems' recommendation systems. We tried to include diverse online systems. When we found multiple papers on a system we chose only the latest, unless others differed in strategies for efficiencies.

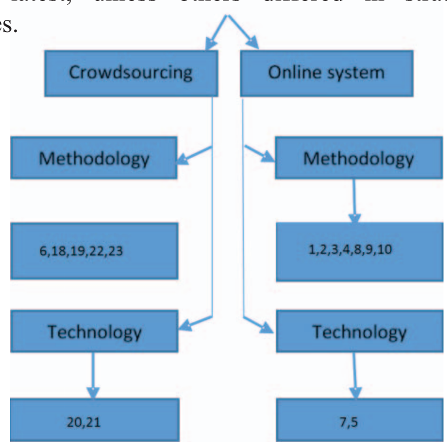

Figure 1 CLASSIFICATION SCHEME

A.1 Methodology papers:

Hwang et al. [1] built a recommendation system in e-commerce applications using the hybrid approach. The main contribution is employing the Genetic Algorithm (GA) to learn the personal preferences of customers.

1- Product features, customer transaction history, and customers' general information.
2- Use product profiles, customer transactions, and customers' general information to build user preferable profiles. Select the most similar neighbors based on the user's preferable profile. Recommend products for the customer based on neighbors' selections.
3- They only satisfy one stakeholder who is the user.
4- A product profile is presented by several features. The customer's product preference profile is built from the customer transaction data and the product profile. The GA is applied to find the feature weighting for a customer. Products are recommended to users by using a collaborative filtering approach.
5- The evaluation is measured by a 5-fold cross-validation approach and it uses the precision metric, recall metric, and F1-measure metric using a real dataset.
6- Not scalable.
7- We are not sure about how the GA is effective in a real time system. There is no experimentation with a real life system and no solution for the new-user problem.

Lin et al. [2] built a news recommendation system by using user-based collaborative approach. The main contribution is utilizing experts' opinions to solve the cold start problem.

1- The reader's history and the reader's implicit rating--which reflects if the reader opens the story or not--are used.
2- From the reader history, find his preference profile. Find experts who influence the reader. Incorporate the reader's preferable factors from preference profile with the experts' opinion to recommend news stories to the reader.
3- They only satisfy one stakeholder who is the news reader.
4- User interest is represented by preferable entities. Then they use it to create user-entity matrix. They use users' news story access history to create a user-story matrix, which is sparse due to the large amount of news stories. To solve that, they predict the missing story rating by using a user-entity matrix. Moreover, they find experts based on the reading time order. Finally, they recommend news stories based on all of the above.
5- The evaluation was measured by comparing the proposed algorithm with three recommendation algorithms using real dataset.
6- Scalable.
7- There is no experimentation with a real life system.

Li et al. [3] built a recommendation system for e-commerce website using the collaborative filtering approach. The main contribution is the design of a kernel-based machine learning approach that assesses individuals' similarities.

1- All the user information is in social networks.
2- By deploying different social network theories, find similar users based on the users' behaviors in the social network. Recommend products to users based on a collaborative filtering approach.
3- This study satisfies one stakeholder who is the user.
4- A kernel-based machine-learning problem is based on social theory to assess individuals' similarities and then recommend products to users based on similar users' choices.
5- The evaluation is measured by comparing the proposed algorithm with a trust-based and collaborative filtering approach by adopting the root mean square error measurement using a real dataset.
6- Scalable.
7- There is no experimentation with a real life system, they neglect the content-based approach in order to recommend more accurately. Moreover, there is no solution for the cold start problem.

Meehan et al. [8] proposed a tourism recommendation using a hybrid approach. The main contribution is adding the Social Media Sentiment factor. They solved the cold start problem by recommending attractions based on the user's current state.

1- The user's location, current weather, current time, Social Media Sentiment--which is positive or negative tweets--as well as personalized information are used.
2- Use user location, time, and current weather to properly recommend to the current state. Use tweets about certain

attractions from a tweeter to sort the priority for each attraction. Combine aforementioned factors to produce appropriate recommendations.

3- They only satisfy one stakeholder who is the user.
4- Extract user-personalized information from social media if it is available; otherwise, ask the user to fill out personal information. They use GPA, GSM, and Wi-Fi to get the accurate location and WorldWeatherOnline API to get the weather information. They use Alchemy API to analyze tweets if it is a positive, negative, or neutral tweet. Then, an artificial neural network could denote each factor as a node and decide a proper weight for each factor. Finally, they list the recommended attractions.
5- The study is still in progress without evaluation.
6- Scalable.
7- There is no detailed explanation or experimentation.

Rong et al. [9] proposed a recommendation system for an e-commerce website using the user-based collaborative approach. The main contribution is solving the cold start problem.

1- Rating history for each user.
2- From users' rating history, find the similarities between a target user and other users to predict the missing elements' rating for the target user.
3- They only satisfy one stakeholder who is the news reader.
4- The system uses a rating matrix to find the similarities between a target user and the other users. For a new user, the system uses an undirected, weighted bipartite graph to find similar users. This graph has two sets of vertices: users and items. They define a random walk to find the similar users using the Monte Carlo algorithm. There are two types of walk: from a user to a previously rated item and from an item to another user who has the most similar rating for that item. Finding the similarities between users is a pre-computed process.
5- The evaluation was measured by using 4-fold cross validation on a real dataset.
6- Scalable.
7- There is no experimentation with a real life system. They did not use a content-based approach as a factor to recommend more accurately and overcome the new-item problem.

Feng et al. [10] proposed a general recommendation system. The main contribution is enhancing the item-based collaborative filtering approach by using K-means clustering.

1- The rating history for each user is used.
2- From the user's rating history, find the similarity between items. Recommend items that meet the user's preferable attributes. Overcome the scalability barrier by using collaborative filtering approach with K-means clustering.
3- They only satisfy one stakeholder who is the news reader.
4- They generate an item-based collaborative result. Then, they generate a K-means clustering result from the collaborative filtering result. Finally, they combine results by using a Simulated Annealing algorithm.
5- The evaluation assessments used a real dataset, the average payoff value, the standard deviation, the precision rate, the recall rate, and the running time of the algorithm.
6- Scalable.
7- There is no experimentation with a real life system.

Dror et al. [4] proposed a recommendation system for Yahoo! Answers using a hybrid approach. The main contribution is using a multi-channel recommender mechanism.

1- Question attributes, which include three classifications: textual, categories and user IDs who interact with the question, were used. Additionally, user properties that include the three aforementioned classifications for each user from the questions answered before plus users' explicit preferences attributes were used.
2- By using question user attributes, predict which users will answer a question. Handle it as a classification problem. Train a Gradient Boosted Decision Trees classifier using logistic loss.
3- It addressed the three stakeholders: the answerer, the asker, and the service provider. This satisfaction is met by default when recommending related questions to the answerer.
4- They proposed a Multi-Channel Recommender model that maps questions and users to their attributes. The question attribute is described by a matrix. User attributes are driven from question attributes that the user has interacted with as several channels. Each channel describes the nature of interaction. Pairing each question attribute with each user attribute creates multiple features, which are used by a classifier to match between users and questions.
5- The evaluation is measured by calculating the accuracy and the Area Under ROC Curve (AUC) on a large-scale dataset.
6- Scalable.
7- There is no experimentation with a real life system.

A.2 Technology Papers:

Davidson et al. [7] proposed a recommendation system for YouTube using a content-based approach. The main contribution is balancing user's specifications and the videos diversity.

1- Watched videos' attributes and user activity data are used.
2- From users' histories, construct a seed set for each user that contains a set of the watched videos. Using the seed set, build a candidate set of videos that fit the user's preferable attributes and meet the diverse condition to recommend it to the user.
3- They only satisfy one stakeholder who is the user.

4- Map each video in the seed set to related videos using co-visitation counts to have the candidate set. Then, get the related videos to each video in the set ranked by their co-visitation count. Then, expand the candidate set by taking a limited transitive closure over the related videos to make the candidate set more diverse.
5- The system works as a feature on YouTube's home page using a pre-computation approach. The data set is updated several times per day. Comparing the recommendation system with other algorithms, the recommended videos were more watched than other video lists.
6- Scalable.
7- The pre-computing generation will cause a delay between generating the recommendation and producing it to the user. The delay will not be severe due to several updates. There is no solution for the new-user problem. Still the recommended videos will be narrower than if collaborative filtering were used.

Cosley et al. [5] presents a recommendation system for Wikipedia using the hybrid approach. The main contribution is the experiment with a real life system.

1- Users' editing history was used.
2- Using the user editing history, find similar articles. Use the explicit connections through links to find related articles, find similarities between users' histories to recommend articles for similar users.
3- It addressed the three stakeholders: the answerer, asker, and service provider. This satisfaction is met by default when recommending more related questions to the answerer.
4- A Jaccard metric for set similarity between profiles, SQL quires to measure text similarity, and explicit connections through links are used.
5- It is implemented by building a recommendation tool for real communities with real users in Wikipedia's website using MySQL 4.1's built-in tool.
6- Scalable.
7- There is no solution for the new-user problem

B. Crowdsourcing Recommendation Papers

In this section, we explore crowdsourcing recommendation papers. We searched Google Scholar for recommender systems in crowdsourcing, recommendation system crowdsourcing, and Task matching. Moreover, in each paper, we looked over the references to find related studies.

B.1 Methodology papers:

Yuen et al. [6] proposed a recommendation system for Amazon Mechanical Turk (Mturk) using the matrix factorization approach. The main contribution is adding a worker task searching history.

1- Task features such as title and time allotted were used. A worker's performance history features such as number of browsed tasks was used. Five worker task searching history features were used: 1) a did not browse the task; 2) browsed the task; 3) worked on the task; 4) completed the task; 5) and accepted the task.
2- By using a worker's performance history and a worker's task searching history, predict the missing values in the worker task matrix.
3- They only satisfy one stakeholder who is the worker.
4- It is solved by using Probabilistic Matrix Factorization to recover the worker-task matrix.
5- To prove the influence of the worker's search history on selecting new tasks, they post a survey as a task for 100 workers.
6- Scalable.
7- There is no experimentation with a real life system. No solution for cold start problem.

Ambati et al. [18] proposed a recommendation system for crowdsourcing platform using a content-based approach. The main contribution is cooperating worker performance and interests to satisfy the requester.

1- User profiles, explicit worker feedback, implicit worker feedback--that is, user-task interaction and task details--were used. Moreover, they used the requester feedback.
2- Use workers' profile information, workers' explicit feedback, workers' implicit feedback, task details, and requesters' feedback to learn the user's task preference.
3- They only satisfy one stakeholder who is the worker.
4- Each task is represented by a set of weighted features. They use Bag-of-Words or classification-based approach to learn the user's task preferences. Then, rank the available tasks to recommend the top few tasks.
5- The evaluation used a real dataset. They use half of the worker's interests to train the system. Then, they rank the other half of the interests. If the recommended task has been completed by a worker that confirms the system effectiveness.
6- Scalable.
7- There is no experimentation with a real life system. They used a content-based approach, which has several limitations as we mentioned previously. Moreover, there is no solution for the cold start problem.

Yuen et al. [19] proposed a task recommendation system for a crowdsourcing platform using a content-based approach. The main contribution is to assist workers to find their suitable tasks.

1- The worker's performance record, and task acceptance rate are used to infer the worker performance in each category.
2- Use the worker's performance records and the worker's task acceptance rates to rank all the available tasks based

on the best match for the worker's interest and performance.

3- They only satisfy one stakeholder who is the worker.
4- Each worker's performance record has four values: 1) task acceptance rate in each category; 2) task category preference score; 3) reward preference score; 4) time allotted preference score. Based on these values, rate all the available tasks. Then list all the available tasks based on the best match.
5- The evaluation was made by applying case study. Moreover, they used Mean Absolute Error to compare the task's user rate from the experiment and the task's user rate that generated by the proposed algorithm.
6- Not scalable.
7- There is no experimentation with a real life system. They used only a content based approach which has major limitations. There is no solution for the cold start problem.

Lin et al. [22] proposed a recommendation system for crowdsourcing using matrix factorization collaborative approach. The main contribution is incorporating negative implicit feedback.

1- User history, which contains user-task interaction is used.
2- Use user history to infer the task's positive or negative rate. Then, recommend the tasks with the highest rating value.
3- They only satisfy one stakeholder who is the worker.
4- The positive feedback has been assessed based on how many times the worker has performed that kind of task. The negative feedback based on the task availability. Then based on the positive and negative rating, recommend tasks by multiplying each predicted rate with task availability in the training set to produce predicted throughputs, and sort tasks accordingly.
5- The evaluation used a real dataset. They compare the proposed system with two other approaches which are the neighbor based approach and the task popularity based approach.
6- Scalable.
7- There is no experimentation with a real life system. Even though collaborating filtering has been proven in several studies to outperform content based, the hybrid approach could outperform both approaches. There is no solution for the cold start problem.

Yuen et al. [23] proposed a recommendation system in crowdsourcing using the matrix factorization collaborative approach. The main contribution is considering the dynamic scenarios for the new worker and the new task.

1- Worker history and tasks' categories are used.
2- Use the worker's history to infer the worker's preferable tasks. From the task, extract the task category. Use worker's preferred tasks to extract worker's preferable categories.
3- They only satisfy one stakeholder who is the worker.
4- The values in the worker-task preference matrix indicate the user's rating for that task. The rating inferred from the worker-task interaction, which ranges from 0-5 as described in the study [6]. Then, they apply matrix factorization to generate worker-categories matrix, and tasks-categories matrix. Then, they recommend tasks that match the user's categories interest.
5- The evaluation was by using the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). They compare their approach with the state of the art approach.
6- Scalable.
7- There is no experimentation with a real life system.

New user and new item that belong to new category is still an issue.

B.2 Technology papers:

Basak et al.'s [20] main contribution is building a framework to test different recommendation algorithms for crowdsourcing.

1- Worker-modeling module, a task-modeling module, and a task recommendation module are used.
2- This paper presents a framework to facilitate recommendation algorithms test for crowdsourcing.
3- The paper did not add a new methodology.
4- The framework has task creation, and management interface to enable the researcher to control the framework. There are three main modules: 1) worker-modeling module; 2) task-modeling module; 3) task recommendation module. Workers and tasks can be modeled using different properties.
5- To test the framework they implement an experiment for three algorithms. 1) feature-independent. By implementing user-item matrix. 2) feature-based. By implementing user-feature matrix. 3) hybrid.
6- It supports a wide variety of recommendation algorithms.
7- The paper did not add any new methodology.

Difallah et al. [21] proposed a recommendation system for crowdsourcing using content based approach. The main contribution is the automated pushing mechanism.

1- User history, which contains liked Facebook pages and completed tasks, is used. Hits description is also used.
2- Use user history information to build the worker's profile that contains the worker's skills and interests. For each submitted task, decompose several micro-task and assign it as a hit with a specified monetary reward. Each liked page linked to an entity in the Linked Open Data (LOD) cloud to categorize them. Then match users with hits based on the similarities between them.
3- They only satisfy one stakeholder who is the worker.

4- Each hit has: a textual description, data field, set of candidate answers, and a list of Facebook categories. Each worker profile is represented by two factors: the liked pages categories and the completed task categories. Then for each hit, rank all the workers based on the matching between the hit and the user. Finally, post the hit to the top matching users.
5- They implement a Facebook App called OpenTurk to push hits to the Facebook users as described above.
6- Scalable.
7- They use only content based approach. No solution for cold start problem.

V. CONCLUSIONS

Crowdsourcing systems have three stakeholders: a worker, a requester, and a service provider. The requester posts the task to the crowd. Workers can accept a multiple number of tasks to get more monetary rewards. The service provider's role is to match workers with tasks accurately in order to aggregate more commission. However, if a worker gets a list of related recommended tasks and works on a large number of tasks at the same time, it could decrease the solution efficiency. As an alternative, part of these tasks could be assigned to less experienced workers who have more time, which could increase the solution efficiency. In another scenario, if we recommend tasks to the most related worker, the requester target may be addressed. However, the worker may get busy processing low monetary tasks and miss some higher monetary reward tasks.

Due to the infancy of the field with limited work [6, 18, 19, 22, 23, 20, and 21] on recommendation systems for crowdsourcing, we reviewed recommendation systems for other applications. Our research shows that most of the recommendation systems address one stakeholder. However, in some systems, satisfying one stakeholder leads to holistic satisfaction [4, and 5]. Ranking of all tasks to recommend, based on the loose conditions of worker history, interest, performance and requester's feedback, has a scalability problem.

Designing a recommendation system that achieves stakeholders' gratification would be a great opportunity for effective crowdsourcing. Moreover, we did not find any work that uses a neighbor-based collaborative filtering approach or a hyped approach, which could outperform the existing methodology.

REFERENCES

[1] Hwang, C. S., Su, Y. C., & Tseng, K. C. (2010). Using genetic algorithms for personalized recommendation. In *Computational Collective Intelligence. Technologies and Applications* (pp. 104-112). Springer Berlin Heidelberg.

[2] Lin, C., Xie, R., Li, L., Huang, Z., & Li, T. (2012, October). Premise: Personalized news recommendation via implicit social experts. In *Proceedings of the 21st ACM international conference on Information and knowledge management* (pp. 1607-1611). ACM.

[3] Li, X., Wang, M., & Liang, T. P. (2014). A multi-theoretical kernel-based approach to social network-based recommendation. *Decision Support Systems*.

[4] Dror, G., Koren, Y., Maarek, Y., & Szpektor, I. (2011, August). I want to answer; who has a question?: Yahoo! answers recommender system. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1109-1117). ACM.

[5] Cosley, D., Frankowski, D., Terveen, L., & Riedl, J. (2007, January). SuggestBot: using intelligent task routing to help people find work in wikipedia. In *Proceedings of the 12th international conference on Intelligent user interfaces* (pp. 32-41). ACM.

[6] Yuen, M. C., King, I., & Leung, K. S. (2012, August). Task recommendation in crowdsourcing systems. In *Proceedings of the First International Workshop on Crowdsourcing and Data Mining* (pp. 22-26). ACM.

[7] Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., ... & Sampath, D. (2010, September). The YouTube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems* (pp. 293-296). ACM.

[8] Meehan, K., Lunney, T., Curran, K., & McCaughey, A. (2013, March). Context-aware intelligent recommendation system for tourism. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on* (pp. 328-331). IEEE.

[9] Rong, Y., Wen, X., & Cheng, H. (2014, April). A Monte Carlo algorithm for cold start recommendation. In *Proceedings of the 23rd international conference on World wide web* (pp. 327-336). International World Wide Web Conferences Steering Committee.

[10] Feng, Z. M., & Su, Y. D. (2013). Application of Using Simulated Annealing to Combine Clustering with Collaborative Filtering for Item Recommendation.*Applied Mechanics and Materials*, *347*, 2747-2751.

[11] Zhu, H., Chen, E., & Cao, H. (2011). Finding experts in tag based knowledge sharing communities. In *Knowledge Science, Engineering and Management*(pp. 183-195). Springer Berlin Heidelberg.

[12] Geiger, D., & Schader, M. (2014). Personalized task recommendation in crowdsourcing information systems—Current state of the art. *Decision Support Systems*.

[13] N. Kaufmann, T.Schulze, D.Veit, More than fun andmoney. Worker motivation in crowdsourcing — astudy on mechanical turk, 17th Am. Conf. Inf.Syst.,Detroit, MI, 2011.

[14] T. Schulze,S.Krug, M. Schader,Workers' task choiceincrowdsourcingand human computation markets, 33rd Int. Conf. Inf. Syst., Orlando, USA, 2012.

[15] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, *17*(6), 734-749.

[16] Lee, S., Yang, J., & Park, S. Y. (2004, January). Discovery of hidden similarity on collaborative filtering to overcome sparsity problem. In Discovery Science(pp. 396-402). Springer Berlin Heidelberg.

[17] Whitla, P. (2009). Crowdsourcing and its application in marketing activities.Contemporary Management Research, 5(1).

[18] Ambati, V., Vogel, S., & Carbonell, J. G. (2011, August). Towards Task Recommendation in Micro-Task Markets. In *Human Computation* (pp. 1-4).

[19] Yuen, M. C., King, I., & Leung, K. S. (2011, October). Task matching in crowdsourcing. In Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing (pp. 409-412). IEEE.

[20] Basak, D., Loni, B., & Bozzon, A. A Platform for Task Recommendation in Human Computation.

[21] Difallah, D. E., Demartini, G., & Cudré-Mauroux, P. (2013, May). Pick-a-crowd: tell me what you like, and i'll tell you what to do. In *Proceedings of the 22nd international conference on World Wide Web* (pp. 367-374). International World Wide Web Conferences Steering Committee.

[22] Lin, C. H., Kamar, E., & Horvitz, E. (2014). Signals in the Silence: Models of Implicit Feedback in a Recommendation System for Crowdsourcing.

[23] Yuen, M. C., King, I., & Leung, K. S. (2014). TaskRec: A Task Recommendation Framework in Crowdsourcing Systems. Neural Processing Letters, 1-16.