Towards Crowd-Assisted Data Mining

Sai R. Gouravajhala
Computer Science & Engineering
University of Michigan, Ann Arbor
sairohit@umich.edu

Danai Koutra
Computer Science & Engineering
University of Michigan, Ann Arbor
dkoutra@umich.edu

Walter S. Lasecki
Computer Science & Engineering
and School of Information
University of Michigan, Ann Arbor
wlasecki@umich.edu

Abstract
Mining massive datasets can benefit from human input, but current approaches require making tradeoffs between overburdening end users or under-informing the system – algorithms become more accurate given more training data, but requiring more exemplars takes significant user effort. In this paper, we suggest an approach that engages non-expert and semi-expert crowds as a supporting "interface layer" between end users and data mining systems. Leveraging human intelligence will allow systems to answer new types of queries (e.g., vague or subjective ones) and generate richer example sets for user-specified patterns.

Using crowdsourcing to parallelize this task makes it possible to provide training data to the system in nearly real-time. This allows the system to learn from crowd-generated examples of user-provided instances within the span of a single query. The user can also post follow-up queries to iteratively refine results. By maintaining an ongoing context for these interactions, we can make the query-respond-refine process resemble a "conversational" interaction between user and system, helping make data analysis more approachable to non-experts.

Author Keywords
crowdsourcing; data mining; conversational system

ACM Classification Keywords
H.5.m [Information interfaces and presentation]: Misc.

Introduction
Human-in-the-loop data mining allows end users to help train scalable algorithms to suit their needs [3]. Unfortunately, current approaches require trading off between overburdening end users and under-informing the system. In order to generate large sets of training data that can better inform the system's underlying algorithms, significant, and often infeasible, amount of effort is required from users. If training is needed to accurately answer an active query, users must provide exemplars before proceeding with their analysis. Augmenting data analysis systems with human intelligence can allow them to (potentially) answer any human-understandable question about data, including vague or subjective ones, such as "find me something interesting in this dataset." Moreover, since people are adept at one-shot learning [10], they can be asked to generate and annotate example sets for user-specified queries and patterns. By using the parallelism of crowds, we can find patterns far more quickly than any individual can.

We can combine these approaches with existing methods for scalable data mining tasks, such as anomaly detection [1], summarization [9], clustering [18], and more.

Related Work
Amershi et al. [2] review the role of humans in interactive machine learning (IML) and outline challenges faced by the field, one of which involves the role of crowds. Work on amplifying community content creation by Hoffmann et al. [6], and the work on Galaxy Zoo by Kamar et al. [7] also falls in this sphere. On the crowdsourcing side, Lasecki and Bigham [12] propose a system that allows crowd workers to keep a collective memory. Furthermore, Cheng and Bernstein [4] show that hybrid machine learning classifiers that aggregate crowd features outperform classifiers that use only crowd-nominated or machine-extracted features.

Prior work has shown that crowds can complete continuous tasks in real-time [16]. Apparition [15], probably the single most related system, is a crowd-powered system for interactively generating functional interface prototypes from sketch and spoken natural language. By focusing worker efforts, it has been shown that the level of expertise needed to contribute to a task can be reduced [11], or taught via micro-training sessions [13]. This means that we can use crowd workers who have knowledge of data analysis, but not the specific domain (semi-experts), or even workers with no specific prior knowledge (non-experts).

Advantages of the Proposed Approach
Our proposed crowd-powered architecture allows us to use *natural language* as the basis for this interaction, which we hope will yield a novel type of flexible data analytics system that maximizes performance from both people (expert and non-expert alike) and automated systems. Users may construct human-understandable queries, even if algorithms may not understand, because the crowd can clarify with examples. Users can also construct queries where they describe patterns they wish to find. The query-respond-refine process allows the crowd's insights to become an integral part of the data analysis workflow. As a result, workers are more than cogs in the machine. Instead, the system integrates and coordinates different contributor roles to enable more effective data analysis.

Learning New Lexicons
As the crowd annotates regions of interest, the system can learn from the crowd's descriptions. The benefit of learning this lexicon is that the system can now map from the natu-

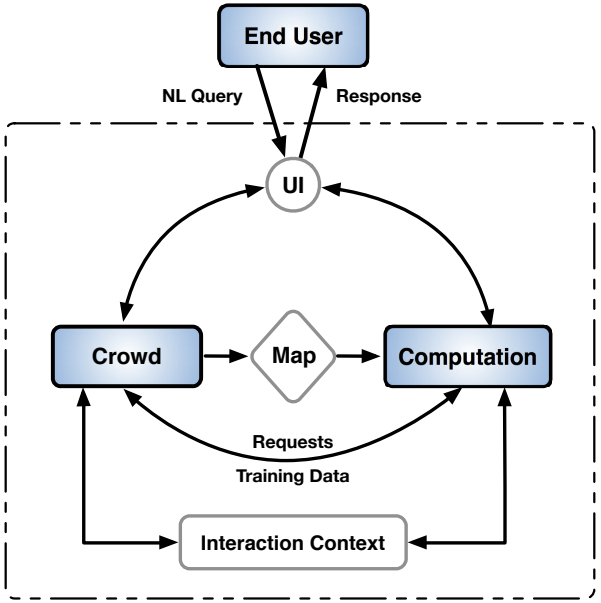

Figure 1: The system comprises all the elements inside the dotted rectangle. The End User uses natural language queries to interact with the system. The Crowd helps the End User with the data analysis by supporting vague or subjective queries. The UI provides the crowd with analysis tools.

ral language descriptors to output attributes, such as filters, sort functions, actions to take on the data, and best-fit algorithms. These mappings allow the system to automate the handling of natural language queries over time. This iterative learning process is similar to a conversational interaction [17, 19].

Interaction Context

Making available the ongoing interaction context to both the crowd and the data analysis can help the crowd get a better idea of what was queried before, and how it is related to the current query. Similarly, the data mining system can use this context to better select models or model parameters. This is akin to context or intent in information retrieval work, and social translucence [5] in digital systems.

Using this interaction history, we can also learn more information about the lineage of a query result. For example, going back in the conversation and constructing the chain of queries and intermediate results that directly led to the final result can help attribution [20]. This aids both the human's and the system's understanding of how a particular query is answered.

Example System Architecture

Our proposed system architecture is shown in Figure 1. The system can route the user's query to the crowd, to a computational module, or to both. Crowd workers are provided an interface for viewing large datasets and attributes (e.g., Perseus [8], a large-scale graph mining and visualization tool). Once the crowd generates a response, the system integrates the findings (e.g., maps the vocabulary in the annotations to data features and updates its models) and returns a result that is the composite of the crowd's responses and the system's guesses after re-training. At any point, the computation module can also request more training data from the crowd if needed. Both modules can curate and refer to the interaction context when completing tasks.

System Considerations and Challenges

Implementing a crowd-powered data mining system will require considering several key points:

- To facilitate a distributed discovery process, we need to be able to train non-expert workers on data analysis tools and techniques.
- To understand "one-shot" examples used by end users to specify patterns, the crowd must know enough about the domain or analysis tools to interpret the user's context and utterance.

- Given the potential variability of crowds, the problem of efficiently ensuring consistent, reliable explanations and examples in large data sets is critical.
- To avoid overwhelming non-experts, we need to develop an interface that accommodates workers of differing skill levels.
- Dividing datasets in a manner that preserves meaning for a given query is necessary to enable groups of crowd workers to operate in parallel.
- If there is one dataset for which we develop a lexicon, a challenge remains regarding how can we have the system identify mappings from this particular lexicon to another one for a different dataset.
- The generalizability of crowds and training across multiple fields and applications will significantly impact how flexible these systems are.
- For privacy-sensitive datasets (e.g., healthcare data), ways are needed to properly anonymize parts of the data without adversely impacting the crowd's ability to generate accurate responses [14].

Conclusion

We suggest that we can engage crowds to provide input and insights as part of an "intermediate layer" between end users and data mining algorithms. This can help make interactions more fluid, and help more effectively leverage existing data mining algorithms. We outlined some key advantages afforded by using crowds in this way and highlight important system design considerations that will likely arise in such systems. We believe this work can spark interesting discussions with the workshop's attendees regarding future directions and related work.

References

[1] Leman Akoglu, Hanghang Tong, and Danai Koutra. 2014. Graph-based Anomaly Detection and Description: A Survey. *Data Mining and Knowledge Discovery (DAMI)* (April 2014).

[2] Saleema Amershi, Maya Cakmak, W Bradley Knox, and Todd Kulesza. 2014. Power to the people: The role of humans in interactive machine learning. *AI Magazine* 35, 4 (2014), 105–120.

[3] Duen Horng Chau, Aniket Kittur, Jason I. Hong, and Christos Faloutsos. 2011. Apolo: Making Sense of Large Network Data by Combining Rich User Interaction and Machine Learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 167–176.

[4] Justin Cheng and Michael S Bernstein. 2015. Flock: Hybrid crowd-machine learning classifiers. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. ACM, 600–611.

[5] Thomas Erickson and Wendy A Kellogg. 2000. Social translucence: an approach to designing systems that support social processes. *ACM transactions on computer-human interaction (TOCHI)* 7, 1 (2000), 59–83.

[6] Raphael Hoffmann, Saleema Amershi, Kayur Patel, Fei Wu, James Fogarty, and Daniel S Weld. 2009. Amplifying community content creation with mixed initiative information extraction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1849–1858.

[7] Ece Kamar, Severin Hacker, and Eric Horvitz. 2012. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and*

Multiagent Systems-Volume 1. International Foundation for Autonomous Agents and Multiagent Systems, 467–474.

[8] Danai Koutra, Di Jin, Yuanchi Ning, and Christos Faloutsos. 2015. Perseus: an interactive large-scale graph mining and visualization tool. *Proceedings of the VLDB Endowment* 8, 12 (2015), 1924–1927.

[9] Danai Koutra, U Kang, Jilles Vreeken, and Christos Faloutsos. 2014. VoG: Summarizing and understanding large graphs. In *Proceedings of the SIAM International Conference on Data Mining (SDM), Philadelphia, PA*. SIAM.

[10] Brenden M Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua B Tenenbaum. 2011. One shot learning of simple visual concepts. In *Proceedings of the 33rd Annual Conference of the Cognitive Science Society*, Vol. 172. 2.

[11] Walter Lasecki, Christopher Miller, Adam Sadilek, Andrew Abumoussa, Donato Borrello, Raja Kushalnagar, and Jeffrey Bigham. 2012. Real-time Captioning by Groups of Non-experts. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, New York, NY, USA, 23–34.

[12] Walter S. Lasecki and Jeffrey P. Bigham. 2013. Automated Support for Collective Memory of Conversational Interactions. In *Human Computation Works-in-Progress*.

[13] Walter S Lasecki, Mitchell Gordon, Danai Koutra, Malte F Jung, Steven P Dow, and Jeffrey P Bigham. 2014. Glance: Rapidly coding behavioral video with the crowd. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 551–562.

[14] Walter S. Lasecki, Mitchell Gordon, Winnie Leung, Ellen Lim, Jeffrey P. Bigham, and Steven P. Dow. 2015a. Exploring Privacy and Accuracy Trade-Offs in Crowdsourced Behavioral Video Coding. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 1945–1954.

[15] Walter S Lasecki, Juho Kim, Nicholas Rafter, Onkur Sen, Jeffrey P Bigham, and Michael S Bernstein. 2015b. Apparition: Crowdsourced User Interfaces That Come To Life As You Sketch Them. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 1925–1934.

[16] Walter S. Lasecki, Kyle I. Murray, Samuel White, Robert C. Miller, and Jeffrey P. Bigham. 2011. Real-time Crowd Control of Existing Interfaces. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, New York, NY, USA, 23–32.

[17] Walter S. Lasecki, Rachel Wesley, Jeffrey Nichols, Anand Kulkarni, James F. Allen, and Jeffrey P. Bigham. 2013. Chorus: A Crowd-powered Conversational Assistant. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM, New York, NY, USA, 151–162.

[18] Mark EJ Newman. 2006. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences* 103, 23 (2006), 8577–8582.

[19] Brent Rossen and Benjamin Lok. 2012. A Crowdsourcing Method to Develop Virtual Human Conversational Agents. *Int. J. Hum.-Comput. Stud.* 70, 4 (April 2012), 301–319.

[20] Neil Shah, Danai Koutra, Tianmin Zou, Brian Gallagher, and Christos Faloutsos. 2015. TimeCrunch: Interpretable Dynamic Graph Summarization. In *Proceedings of the 21st ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*.