Towards Hybrid Human-Machine Workflow for Natural Language Generation

Neslihan Iskender, Tim Polzehl, Sebastian Möller
Technische Universität Berlin, Quality and Usability Lab
{neslihan.iskender, tim.polzehl1, sebastian.moeller}@tu-berlin.de

Abstract

In recent years, crowdsourcing has gained much attention from researchers to generate data for Natural Language Generation (NLG) tools or to evaluate them. However, the quality of crowdsourced data has been questioned repeatedly because of the complexity of NLG tasks and crowd workers' unknown skills. Moreover, crowdsourcing can also be costly and often not feasible for large-scale data generation or evaluation. To overcome these challenges and leverage the complementary strengths of humans and machine tools, we propose a hybrid human-machine workflow designed explicitly for NLG tasks with real-time quality control mechanisms under budget constraints. This hybrid methodology is a powerful tool for achieving high-quality data while preserving efficiency. By combining human and machine intelligence, the proposed workflow decides dynamically on the next step based on the data from previous steps and given constraints. Our goal is to provide not only the theoretical foundations of the hybrid workflow but also to provide its implementation as open-source in future work.

1 Introduction

With the rapid development of Internet technologies, crowdsourcing has become one of the primary resources to solve tasks such as image tagging, transcribing the text, or digitizing print documents that computers cannot yet solve and need human intelligence (Bernstein et al., 2010; Kittur et al., 2011; Tran-Thanh et al., 2015). Further, the cost and time advantages of crowdsourcing have raised the interest of many NLG researchers to generate corpus or to evaluate the quality of NLG outputs (Callison-Burch, 2009; Zaidan and Callison-Burch, 2011; Falke et al., 2017; Fan et al., 2018; Gao et al., 2018). Despite the increasing popularity of crowdsourcing, the quality of crowdsourced data has been many times questioned because of crowd worker‘s potential inaccuracy and the complexity of NLG tasks. As a solution, a variety of workflow approaches have been proposed with the aim of quality assurance, quality control, or cost optimization (Kamar et al., 2012; Kulkarni et al., 2012; Lin et al., 2012; Dai et al., 2013; Lofi and Maarry, 2014; Tran-Thanh et al., 2015; Goto et al., 2016; Retelny et al., 2017; Chen et al., 2019; Jiang et al., 2020).

However, all of these approaches are neither designed explicitly for the given NLG task nor integrate the NLG tools themselves into the workflow dynamically. Therefore, we propose an automatic hybrid human-machine workflow that decides on the next step (when to use humans and when to use an NLG tool) based on the given constraints and the previous workflow step, optimizing the cost/quality trade-off. With this hybrid dynamic methodology, we aim to collect high-quality data while preserving efficiency. Since this is a work-in-progress, we describe the logic and the theoretical aspects of the workflow in this paper and will provide its complete modeling and practical implementation as open-source in future work.

2 Related Work

Many crowdsourcing platforms provide support for repetitive independent micro-tasks, which can be completed in a short amount of time (Hélouët et al., 2020). However, the recent advances of Internet technologies require human intelligence for more complex tasks. As a solution, crowdsourcing workflows have been introduced to a variety of problems such as taxonomy creation (Chilton et al., 2013), entity resolution (Wang et al., 2012), and complex work (Kittur et al., 2011; Kulkarni et al., 2012). The main focus of these crowdsourcing workflows are cost/quality optimization, task allocation, modeling the incentive mechanism, or modeling the

Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing, pages 1–7
April 20, 2021.

crowd workers' skills (Kamar, 2016).

With artificial intelligence (AI) systems being an important part of our lives, combining crowdsourcing workflows with AI tools, *hybrid intelligence*, promise great potential for improving human-only workflows. Therefore, researchers have developed intelligent hybrid systems for real-time speech transcribing (Kushalnagar et al., 2012; Lasecki and Bigham, 2012; Lasecki et al., 2012, 2013, 2017), clustering data points (Gomes et al., 2011; Tamuz et al., 2011; Heikinheimo and Ukkonen, 2013), forecasting political or economic events (Baron et al., 2014; Mellers et al., 2015; Atanasov et al., 2017) or scheduling conference meetings (André et al., 2013; Kim et al., 2013; Bhardwaj et al., 2014; Chilton et al., 2014). These hybrid workflows have been proven to perform better than human-only and machine-only systems.

However, to this date, there is no hybrid human-machine workflow that combines the human and machine intelligence with quality control mechanisms for crowd workers and with a methodology for cost/quality optimization. Usage of crowdsourcing to NLG has been limited to single crowdsourcing studies for quality evaluation or data labeling for semantic parsing (Wang et al., 2015), information retrieval (Demartini, 2015), translation (Callison-Burch, 2009; Zaidan and Callison-Burch, 2011) and summarization (Falke et al., 2017; Fan et al., 2018; Gao et al., 2018; Iskender et al., 2020), but the hybrid intelligence approach has not been applied in these works. Therefore, we propose to combine the strength of the human-only workflows and NLG tools in the form of a hybrid human-machine workflow with quality control mechanisms. Such an integrative hybrid approach offers great promise for the development of practical applications by achieving high-quality data while preserving efficiency.

3 Hybrid Human-Machine Workflow for NLG

3.1 Research Aim

Our goal is to provide a hybrid human-machine workflow optimizing the cost/quality trade-off and its complete implementation using a workflow engine. First, we will integrate the existing state-of-the-art NLG tools into the workflow to create a hybrid human-machine workflow. Following this, we will model each step in the workflow to increase efficiency in terms of cost/quality trade-off. Based on the model and empirical data, the workflow will decide dynamically on the next step whether to use an NLG tool or humans. Additionally, we will implement this workflow using a workflow engine and provide its implementation as open-source. Such a workflow would be especially beneficial for NLG tools developed for low-resource languages, for which it is harder to acquire available data sets. In other languages, researchers usually need to create the data set from scratch for the specific NLG task with linguistic experts, which is extremely expensive and time-consuming for large-scale datasets.

3.2 Workflow Logic

Figure 1 illustrates the workflow logic. To explain it in detail, we use the summarization task as an example of NLG tasks and demonstrate each step in workflow for this task. The workflow starts with the following inputs to the system: new source document to be summarized, budget and time limit, and expected quality level. Based on these input factors (source text length and domain, budget and time limit), the algorithm in *D0: Creation Method* decides whether the summaries should be created by automatic tools, crowd workers, or experts.

In machine creation, the workflow logic chooses the most applicable summarization algorithm based on the source document characteristics such as language, length, domain, and the number of documents. If crowd creation is chosen, the input factors determine the crowdsourcing task design, such as the required qualification of crowd workers, payment, number of crowd workers and repetition patterns, and task duration. If the workflow decides for the expert creation, the created summary will be stored in the database, and the workflow will end because expert creation is the gold standard in NLG (van der Lee et al., 2019).

After crowd summary creation, there is a quality check for each summary to eliminate obvious cheaters and low-quality answers. This quality check is triggered after each crowd answer, and it works on a single answer basis. If the algorithm determines that the crowd worker is cheating (path *fail*), then the answer will be rejected, and the crowd worker will not be paid. The workflow will go back to *D0* state to decide again about the creation method. If the crowd worker is not cheating (path *success*) or machine summary is created, then the workflow goes to state *D1: Evaluation*

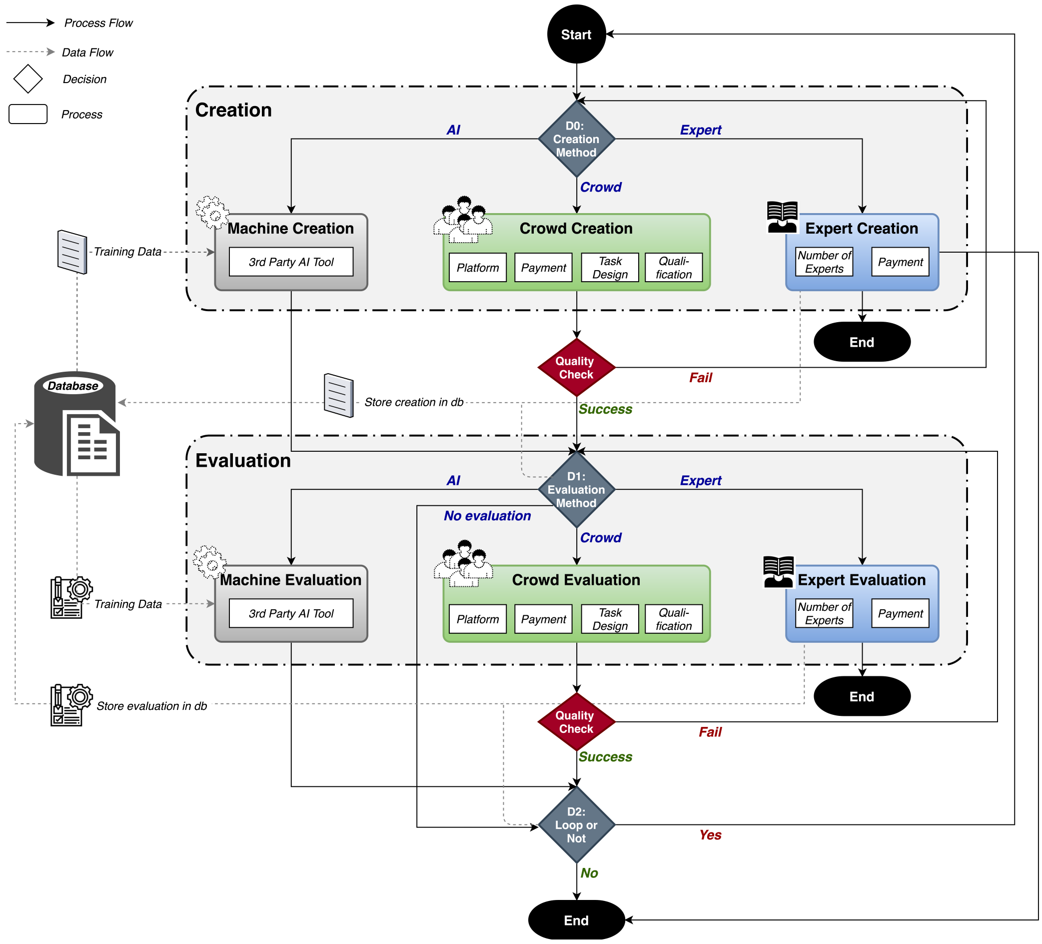

Figure 1: The Logic of the Hybrid Human-Machine Workflow for NLG

Method to decide about the evaluation method. In this state, all the summaries created in the creation part will be sent to the database to be stored. If the workflow decides that it cannot estimate the quality reliably at this early step, it may suggest triggering additional crowd-evaluation.

Analog to the creation part, in the evaluation part, the workflow establishes the most applicable summarization evaluation method based on the input factors from state *D0* and summary characteristics. If the machine creation is chosen, an automatic evaluation tool will evaluate the summary quality. If the algorithm decides for the crowd evaluation, the time, budget, source, and summary characteristics determine the task design, payment, requirements for crowd workers, and the number of crowd workers, and similar to creation, there is again cheater detection step after crowd evaluation. Lastly, in the case of expert evaluation, the evaluations will be directly stored in the database, and the workflow will be ended since the expert evaluation is the gold standard evaluation in NLG.

After successful crowd evaluation or machine evaluation, the workflow reaches the final decision step *D2: Loop or not*. Here, all the evaluation data will be sent to the database to be stored. Based on previous states' information, the workflow algorithm determines if the collected data is satisfying the requirements, e.g., cost and time limit, quality expectation, etc. In the following cases, the workflow will terminate: 1) if the given cost and time budgets are exceeded, or 2) if the quality of collected data satisfies the expected quality. Otherwise, the workflow will go back to *Start* state, and the whole process will be repeated, and results from the current loop serving as (additional) reference or for decisions of *D0*, *D1* and *D2*. After collecting sufficient number of summaries and summary evaluations, the stored data can be used for training summarization tools or improving the existing supervised summarization evaluation metrics.

3.3 Workflow Modeling

We plan to model the logic of the hybrid human-machine workflow as Markov Decision Process (MDP). MDP is defined as a discrete-time stochas-

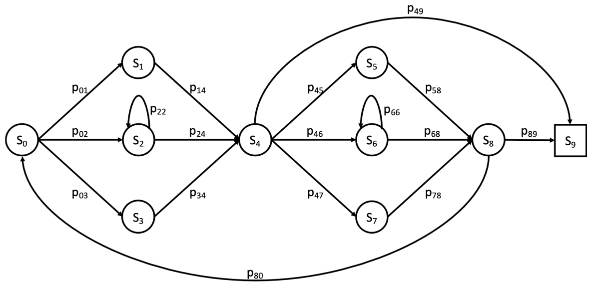

Figure 2: The Modeling of the Hybrid Human-Machine Workflow as a Markov Decision Process

tic control process providing a mathematical framework for modeling decision making in situations where outcomes are partly random and partly under the control of a decision-maker (Feinberg and Shwartz, 2012). The reason for choosing MDP is to model the uncertainty and randomness of workflow added by the humans in the process and subjectivity of NLG tasks. Dai et al. (2013) have already shown that MDPs are useful for optimizing crowdsourcing workflows in terms of cost and quality via dynamic programming.

Figure 2 shows the MDP representation of the hybrid human-machine workflow explained in section 3.2. An MDP is a four-tuple $\langle S, A, T, R \rangle$, where S is a finite set of discrete states (the nodes in figure 2); A is a finite set of all actions (the paths in figure 2); $T : S \times A \times S \rightarrow [0, 1]$ is the transition function; $R : S \times A \rightarrow R$ is the reward for taking an action in a state; $\pi : S \rightarrow A$ is the policy mapping states to actions; and $Q^* - value$ is the value of state action pair (s, a).

The refinement of the transition function, the reward, and $Q^* - value$ will be part of future work. We plan to solve the MDP model by empirically collecting data from workflow applications for several NLG tasks and the Monte Carlo (MC) simulation algorithm repeatedly simulating the trials originating from the *Start*.

3.4 Workflow Implementation

The scientific workflows are generally represented as directed acyclic graphs (DAGs), which illustrate the computational tasks as nodes and the dependencies between them as edges (Liu et al., 2015). The task-driven approach, used by many workflow management engines such as Makeflow (Albrecht et al., 2012), and Pegasus (Deelman et al., 2015), is the traditional approach that relies on triggering tasks when the dependencies are satisfied. As next-generation task-based approach, Airbnb has developed an open-source workflow engine Apache Airflow[1] which can trigger tasks without satisfying dependencies. Another recent approach is triggering workflow by data input and output rather than task dependencies. Popular data-driven workflow engines are Nextflow (Di Tommaso et al., 2017) and Apache Hadoop YARN (Vavilapalli et al., 2013).

Mitchell et al. (2019) did a comparative analysis of common task- and data-driven workflow engines and showed that Apache Airflow suits both task- and data-driven systems with its modular structure and built-in operators. Apache Airflow is written on Python without any other requirement, so the workflow implementation is relatively easy and flexible. With its scheduling feature (each task in the workflow can be scheduled individually), the whole DAG can be triggered periodically, e.g., hourly or daily. Although it is not as robust as data-driven workflow engines, Apache Airflow allows data flow between tasks. Therefore, we plan to implement our hybrid human-machine workflow with Apache Airflow, supporting data flow by external databases.

4 Expected Contributions and Future Work

The current crowdsourcing platforms offer minimal guidance and support on how to interpret the collected data or how to assure quality. With this hybrid dynamic methodology, we aim to overcome this challenge by providing the logic, modeling, and implementation of a hybrid human-machine workflow for NLG with quality control and cost optimization methods. In this way, the data creation and evaluation can be accelerated for many languages leading to the enhancement of multilingual NLG tools. Since many NLG tasks usually require data creation and evaluation steps, the workflow can be adjusted easily to other NLG tasks such as translation, question-answering, or data-to-text.

As future work, after completing the modeling and implementation, we plan to run experiments with the proposed workflow to finalize the workflow decision algorithm and test it empirically. To foster the development of the proposed workflow, we welcome researchers in the NLG community to join our experiments, use the proposed workflow to collect data or evaluate their tool, improve the workflow based on empirical data and serve high-quality results for the researchers.

[1] http://airflow.apache.org/

References

Michael Albrecht, Patrick Donnelly, Peter Bui, and Douglas Thain. 2012. Makeflow: A portable abstraction for data intensive computing on clusters, clouds, and grids. In *Proceedings of the 1st ACM SIGMOD Workshop on Scalable Workflow Execution Engines and Technologies*, SWEET '12, New York, NY, USA. Association for Computing Machinery.

Paul André, Haoqi Zhang, Juho Kim, Lydia Chilton, Steven Dow, and Robert Miller. 2013. Community clustering: Leveraging an academic crowd to form coherent conference sessions. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 1(1).

Pavel Atanasov, Phillip Rescober, Eric Stone, Samuel A. Swift, Emile Servan-Schreiber, Philip Tetlock, Lyle Ungar, and Barbara Mellers. 2017. Distilling the wisdom of crowds: Prediction markets vs. prediction polls. *Manage. Sci.*, 63(3):691–706.

Jonathan Baron, Barbara A Mellers, Philip E Tetlock, Eric Stone, and Lyle H Ungar. 2014. Two reasons to make aggregated probability forecasts more extreme. *Decision Analysis*, 11(2):133–145.

Michael S. Bernstein, Greg Little, Robert C. Miller, Björn Hartmann, Mark S. Ackerman, David R. Karger, David Crowell, and Katrina Panovich. 2010. Soylent: A word processor with a crowd inside. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, page 313–322, New York, NY, USA. Association for Computing Machinery.

Anant Bhardwaj, Juho Kim, Steven Dow, David Karger, Sam Madden, Rob Miller, and Haoqi Zhang. 2014. Attendee-sourcing: Exploring the design space of community-informed conference scheduling. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 2(1).

Chris Callison-Burch. 2009. Fast, cheap, and creative: Evaluating translation quality using Amazon's Mechanical Turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 286–295, Singapore. Association for Computational Linguistics.

Rong Chen, Bo Li, Hu Xing, and Yijing Wang. 2019. Crowdiy: How to design and adapt collaborative crowdsourcing workflows under budget constraints. In *International Conference on Web Engineering*, pages 203–210. Springer.

Lydia B. Chilton, Juho Kim, Paul André, Felicia Cordeiro, James A. Landay, Daniel S. Weld, Steven P. Dow, Robert C. Miller, and Haoqi Zhang. 2014. Frenzy: Collaborative data organization for creating conference sessions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, page 1255–1264, New York, NY, USA. Association for Computing Machinery.

Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. 2013. *Cascade: Crowdsourcing Taxonomy Creation*, page 1999–2008. Association for Computing Machinery, New York, NY, USA.

Peng Dai, Christopher H. Lin, Mausam, and Daniel S. Weld. 2013. Pomdp-based control of workflows for crowdsourcing. *Artif. Intell.*, 202(1):52–85.

Ewa Deelman, Karan Vahi, Gideon Juve, Mats Rynge, Scott Callaghan, Philip J. Maechling, Rajiv Mayani, Weiwei Chen, Rafael Ferreira da Silva, Miron Livny, and Kent Wenger. 2015. Pegasus, a workflow management system for science automation. *Future Gener. Comput. Syst.*, 46(C):17–35.

Gianluca Demartini. 2015. Hybrid human–machine information systems: Challenges and opportunities. *Computer Networks*, 90:5 – 13. Crowdsourcing.

Paolo Di Tommaso, Maria Chatzou, Evan W Floden, Pablo Prieto Barja, Emilio Palumbo, and Cedric Notredame. 2017. Nextflow enables reproducible computational workflows. *Nature biotechnology*, 35(4):316–319.

Tobias Falke, Christian M. Meyer, and Iryna Gurevych. 2017. Concept-map-based multi-document summarization using concept coreference resolution and global importance optimization. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 801–811, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–54, Melbourne, Australia. Association for Computational Linguistics.

Eugene A Feinberg and Adam Shwartz. 2012. *Handbook of Markov decision processes: methods and applications*, volume 40. Springer Science & Business Media.

Yang Gao, Christian M. Meyer, and Iryna Gurevych. 2018. APRIL: Interactively learning to summarise by combining active preference learning and reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4120–4130, Brussels, Belgium. Association for Computational Linguistics.

Ryan Gomes, Peter Welinder, Andreas Krause, and Pietro Perona. 2011. Crowdclustering. In *Advances in Neural Information Processing Systems*, volume 24, pages 558–566. Curran Associates, Inc.

Shinsuke Goto, Toru Ishida, and Donghui Lin. 2016. Understanding crowdsourcing workflow: Modeling and optimizing iterative and parallel processes. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 4(1).

Hannes Heikinheimo and Antti Ukkonen. 2013. The crowd-median algorithm. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 1(1).

Loïc Hélouët, Zoltan Miklos, and Rituraj Singh. 2020. Cost and quality assurance in crowdsourcing workflows. Hal-02964736.

Neslihan Iskender, Tim Polzehl, and Sebastian Möller. 2020. Best practices for crowd-based evaluation of German summarization: Comparing crowd, expert and automatic evaluation. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 164–175, Online. Association for Computational Linguistics.

Youxuan Jiang, Huaiyu Zhu, Jonathan K. Kummerfeld, Yunyao Li, and Walter Lasecki. 2020. A novel workflow for accurately and efficiently crowdsourcing predicate senses and argument labels. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 415–421, Online. Association for Computational Linguistics.

Ece Kamar. 2016. Directions in hybrid intelligence: Complementing ai systems with human intelligence. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 4070–4073. AAAI Press.

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*, AAMAS '12, page 467–474, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Juho Kim, Haoqi Zhang, Paul André, Lydia B. Chilton, Wendy Mackay, Michel Beaudouin-Lafon, Robert C. Miller, and Steven P. Dow. 2013. Cobi: A community-informed conference scheduling tool. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, page 173–182, New York, NY, USA. Association for Computing Machinery.

Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E. Kraut. 2011. Crowdforge: Crowdsourcing complex work. UIST '11, page 43–52, New York, NY, USA. Association for Computing Machinery.

Anand Kulkarni, Matthew Can, and Björn Hartmann. 2012. Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, page 1003–1012, New York, NY, USA. Association for Computing Machinery.

Raja S. Kushalnagar, Walter S. Lasecki, and Jeffrey P. Bigham. 2012. A readability evaluation of real-time crowd captions in the classroom. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '12, page 71–78, New York, NY, USA. Association for Computing Machinery.

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, page 23–34, New York, NY, USA. Association for Computing Machinery.

Walter S. Lasecki and Jeffrey P. Bigham. 2012. Online quality control for real-time crowd captioning. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '12, page 143–150, New York, NY, USA. Association for Computing Machinery.

Walter S. Lasecki, Christopher D. Miller, and Jeffrey P. Bigham. 2013. *Warping Time for More Effective Real-Time Crowdsourcing*, page 2033–2036. Association for Computing Machinery, New York, NY, USA.

Walter S. Lasecki, Christopher D. Miller, Iftekhar Naim, Raja Kushalnagar, Adam Sadilek, Daniel Gildea, and Jeffrey P. Bigham. 2017. Scribe: Deep integration of human and machine intelligence to caption speech in real time. *Commun. ACM*, 60(9):93–100.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan. Association for Computational Linguistics.

Christopher Lin, Mausam Mausam, and Daniel Weld. 2012. Dynamically switching between synergistic workflows for crowdsourcing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 26(1).

Ji Liu, Esther Pacitti, Patrick Valduriez, and Marta Mattoso. 2015. A survey of data-intensive scientific workflow management. *Journal of Grid Computing*, 13(4):457–493.

C. Lofi and K. E. Maarry. 2014. Design patterns for hybrid algorithmic-crowdsourcing workflows. In *2014 IEEE 16th Conference on Business Informatics*, volume 1, pages 1–8.

Barbara Mellers, Eric Stone, Terry Murray, Angela Minster, Nick Rohrbaugh, Michael Bishop, Eva Chen, Joshua Baker, Yuan Hou, Michael Horowitz, et al. 2015. Identifying and cultivating superforecasters as a method of improving probabilistic predictions. *Perspectives on Psychological Science*, 10(3):267–281.

R. Mitchell, L. Pottier, S. Jacobs, R. F. d. Silva, M. Rynge, K. Vahi, and E. Deelman. 2019. Exploration of workflow management systems emerging features from users perspectives. In *2019 IEEE*

International Conference on Big Data (Big Data), pages 4537–4544.

Daniela Retelny, Michael S. Bernstein, and Melissa A. Valentine. 2017. No workflow can ever be enough: How crowdsourcing workflows constrain complex work. *Proc. ACM Hum.-Comput. Interact.*, 1(CSCW).

Omer Tamuz, Ce Liu, Serge Belongie, Ohad Shamir, and Adam Tauman Kalai. 2011. Adaptively learning the crowd kernel. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 673–680, Madison, WI, USA. Omnipress.

Long Tran-Thanh, Trung Dong Huynh, Avi Rosenfeld, Sarvapali Ramchurn, and Nicholas Jennings. 2015. Crowdsourcing complex workflows under budget constraints. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1).

Vinod Kumar Vavilapalli, Arun C. Murthy, Chris Douglas, Sharad Agarwal, Mahadev Konar, Robert Evans, Thomas Graves, Jason Lowe, Hitesh Shah, Siddharth Seth, Bikas Saha, Carlo Curino, Owen O'Malley, Sanjay Radia, Benjamin Reed, and Eric Baldeschwieler. 2013. Apache hadoop yarn: Yet another resource negotiator. In *Proceedings of the 4th Annual Symposium on Cloud Computing*, SOCC '13, New York, NY, USA. Association for Computing Machinery.

Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. 2012. Crowder: Crowdsourcing entity resolution. *Proc. VLDB Endow.*, 5(11):1483–1494.

Yushi Wang, Jonathan Berant, and Percy Liang. 2015. Building a semantic parser overnight. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1332–1342, Beijing, China. Association for Computational Linguistics.

Omar F. Zaidan and Chris Callison-Burch. 2011. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1220–1229, Portland, Oregon, USA. Association for Computational Linguistics.