

# Research Statement

Ting-Hao (Kenneth) Huang

I build future conversational systems that assist people in their everyday lives. Conversation is powerful because the back-and-forth interaction in rich natural language allows people to quickly come to a shared understanding, and thus achieves goals efficiently. Modern intelligent personal assistants illustrate a future in which getting help from a computer is as easy as asking a friend. However, despite decades of research, these systems are still limited in domain, expressiveness, and robustness. My research blends **real-time human computation** with **artificial intelligence** to reliably facilitate human-like conversations. I developed and deployed a crowd-powered conversational assistant, *Chorus*, that users can talk to naturally from day one. My research starts with this deployed working system, and makes it more robust and scalable by incorporating existing Web APIs, by generating executable IF-THEN rules, and most importantly, by learning to automate itself over time.

## Developing and Deploying A Crowd-Powered Conversational Assistant

One core advantage of starting with a working system is that users can talk to it naturally from day one, so we can gather data that could not be collected otherwise. I developed and deployed Chorus<sup>1</sup>, a crowd-powered conversational assistant that can hold open conversations about nearly any topic. When a user initiates a conversation, a group of crowd workers are recruited from Amazon Mechanical Turk and directed to a worker interface allowing them to *propose* responses, *take notes* on important facts, and *vote* on others' responses to identify the optimal responses (Figure 2). By performing these actions

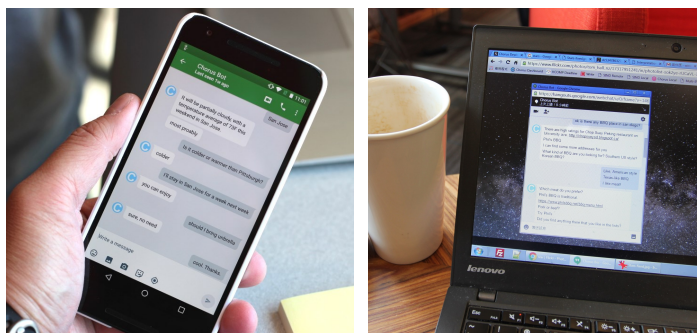


Figure 1: Chorus is a crowd-powered conversational assistant deployed via Google Hangouts, which lets users access it from their computers, phones and smartwatches.

collectively, the crowd is able to converse with the user as a single consistent conversational partner.

Chorus was launched as a Google Hangouts chatbot in May, 2016. Users can talk to Chorus on any of their devices, including smartphones, smartwatches, and desktops at anytime, anywhere (Figure 1). To date, 200+ users have held conversations with Chorus during 1,000+ conversational sessions. The conversations Chorus had with real-world users covered a wide breadth of topics, including weather, travel planning, birthday gift suggestions, relationship consulting, political discussion, and shopping [9]. Multiple challenges emerged during the deployment, including identifying the natural end of conversations, malicious users and workers, and on-demand recruiting [4]. Observations from this deployment not only improve Chorus, but also inform future deployments of low-latency crowd systems in general.

## Turning Existing Web APIs into Automated Dialog Systems

Computers now have access to an incredible amount of information. Web-accessible APIs (Application Programming Interfaces) can be viewed as a gateway to the rich information stored on the Internet. As of November 2017, ProgrammableWeb.com alone contains the description of more than 18,000 APIs. If Chorus is able to exploit the rich information provided by the thousands of available APIs on the web, its scope would be significantly enlarged. However, automatically incorporating Web APIs into a conversational system is a non-trivial task. Despite proposed frameworks to reduce the engineering efforts of

<sup>1</sup>Chorus is available at: <http://TalkingToTheCrowd.org/>



Figure 2: The Chorus interfaces for end users and crowd workers. End users see an interface similar to an instant messenger window, whereas crowd workers see an interface that allows them to see candidate responses, to vote for responses of other workers, and a working memory space. Incentive mechanisms moderate the frequency at which workers are motivated to take different actions (contributing a message, voting for another worker’s message, etc).

developing a dialog system, constructing spoken language interfaces is still a costly endeavor. To tackle these challenges, I introduced *Guardian*, a crowd-powered framework that wraps existing Web APIs into immediately usable conversational agents [10]. Guardian takes as input the Web API and desired task, and the crowd determines the parameters necessary to complete it [7], how to ask for them, and interprets the responses from the API. The system is structured so that, over time, it can learn to take over for the crowd. This hybrid systems approach will help make dialog systems both more general and more robust going forward.

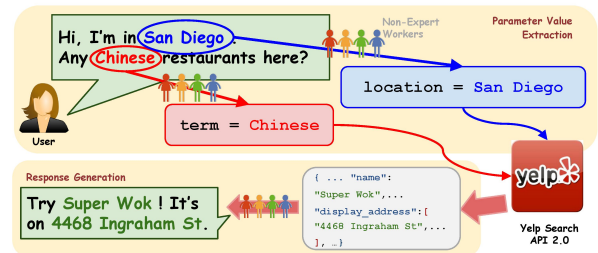


Figure 3: Guardian takes as input the Web API and desired task, and the crowd determines the parameters necessary to complete it, how to ask for them, and interprets the responses from the API.

### Creating IF-THEN Rules via Conversations

Smartphones contain a wealth of applications that could be combined to perform useful tasks and control smart devices around users. If Chorus can *do* something or interact with the user’s environment, it will be able to assist people in addition to providing information. Therefore, I created *InstructableCrowd*, a system that allows end users to create rich, multi-part IF-THEN rules via conversations with the crowd [2]. Users verbally express a problem to crowd workers, who collectively program relevant IF-THEN rules to

help them via conversations. InstructableCrowd allows users to create these rules via voice, on-the-go, and does not require dealing with a complicated interface. Meanwhile, using IF-THEN rules as medium to control devices also protects user’s privacy because the crowd does not have direct access to the user’s devices. InstructableCrowd achieves a comparable average rule accuracy to that of users themselves, and incremental editing on crowd-created rules resulted in an even better performance. InstructableCrowd illustrates how users may converse with a crowd-powered conversational assistant to personalize their increasingly powerful and complicated devices [3].

## A System that Automates Itself Over Time

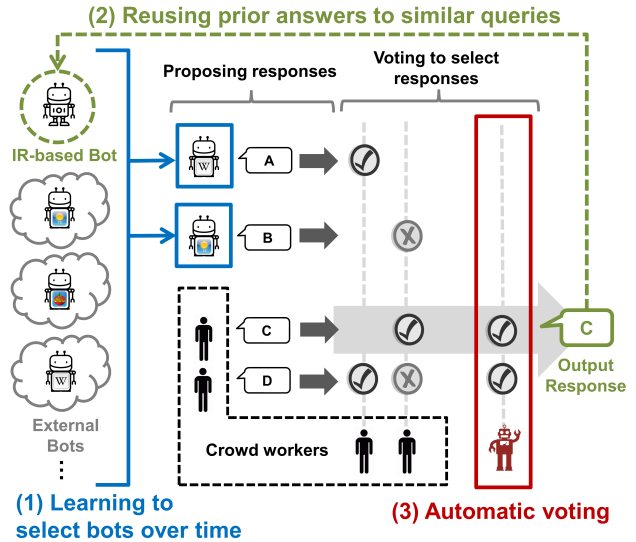
Human involvement enables Chorus to hold long, sophisticated conversations, but does so at the cost of higher response latency and monetary costs. A promising direction is to combine the crowd with automated approaches for high quality, low latency, and low cost solutions. However, transitioning from the crowd to automation has been limited in practice. My Evorus system provides a well-scoped path from crowd-powered robustness to automated speed and frugality [5, 6]. Users chat with Evorus, and it responds. The responses are chosen from suggestions offered by crowd workers and any number of automated systems such as chatterbots, task-oriented dialog systems, or response generation models. Evorus supports increased automation over time in three ways (Figure 4): (i) learning to include responses from chatterbots and task-oriented dialog systems over time, (ii) reusing responses previously generated by the crowd as candidates for new responses, and (iii) gradually reducing the crowd’s role in choosing high-quality responses by partially automating voting. A deployment study showed that Evorus can automate itself without compromising conversation quality. AI-crowd architectures have long been proposed as a way to reduce cost and latency for crowd-powered systems. Evorus demonstrates how automation can be introduced successfully in a deployed system. Its architecture allows other researchers to innovate and compete on improving underlying automated components in the context of a deployed open domain dialog system.

## Research Agenda

My Ph.D. work focused on crowd-powered conversational systems and their extensive possibilities. In the future, I look forward to study these topics with greater depth and larger scale. With my experience in studying and deploying Chorus, I am also excited to explore other crowd-AI systems.

### 1. Deploy Chorus as an Open Research Platform

For the long-term future, Chorus will serve as an open research platform, which has its own growing user base and a group of active crowd workers to operate it, for conducting *live experiments* of technologies. I am excited to pursue the following explorations using the deployed Chorus platform:



**Figure 4: Evorus is a crowd-powered conversational assistant that automates itself over time by (i) learning to include responses from chatterbots and task-oriented dialog systems over time, (ii) reusing past responses, and (iii) gradually reducing the crowd’s role in choosing high-quality responses by partially automating voting.**

- **Chorus Dialog System Challenge:** Testing dialog systems with a *live* system setup is difficult, so majority of them were evaluated off-line. With the deployed Chorus, hosting a shared task that invites different dialog systems to contribute responses and compete with each other (e.g., by response acceptance rate) becomes possible, and can bring new ground and perspective to the community.
- **Coordinating 10,000+ chatbots:** My vision of the future Chorus is a system that fuses 10,000+ response contributors, and developing a sophisticated and robust crowd-AI framework that supports this level of scaling is certainly exciting.
- **Open Chorus API:** It takes a tremendous amount of effort to build a real-time crowd-powered system. To make such technologies easier for other researchers to apply, I will release the *Open Chorus API*. Making Chorus available to the community will encourage more researchers to explore various types of crowd-powered systems, and can benefit the research community as a whole.

## 2. Crowd-Powered Systems on Smart Devices

I imagine a future in which smart devices can perform complex and customized tasks. To do so, human-in-the-loop architectures are essential. Interactive crowd-powered systems can not only be implemented in smartphones or as web applications, but also be applied to smart homes, smart watches [11], voice-enabled devices such as Amazon’s Echo, or even smart cars. Such systems will open up many exciting possibilities and also bring in new challenges, such as the extremely short response time that users expect when talking to a voice-enabled device, hardware design, and users’ privacy concerns.

## 3. Future Crowd-AI Systems

One theme that emerged from my Ph.D. work is using the real-time collaboration between the crowd and AI to create more powerful, robust, and scalable systems. In the future, I will apply the knowledge that I learned from my many projects to develop crowd-AI systems, especially for tasks that are useful to people but still challenging for AI alone, such as visual storytelling [8], which was just introduced to the AI community two years ago; or identifying subtle emotions in messages, which is helpful under many circumstances but hard to automate [12]; or recognizing deaf speech, which today’s speech recognition systems perform poorly but the crowd could improve [1].

## References

- [1] Jeffrey P. Bigham, Raja Kushalnagar, **Ting-Hao K. Huang**, Juan Pablo Flores, and Saiph Savage. 2017. On How Deaf People Might Use Speech to Control Devices. In *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS ’17)*. ACM, New York, NY, USA, 383–384. DOI:<http://dx.doi.org/10.1145/3132525.3134821>
- [2] **Ting-Hao K. Huang**, Amos Azaria, and Jeffrey P Bigham. 2016. InstructableCrowd: Creating IF-THEN Rules via Conversations with the Crowd. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI’16 Late Breaking Work)*. ACM, 1555–1562. [[Best Paper Honorable Mention Award of LBW](#)].
- [3] **Ting-Hao K. Huang**, Amos Azaria, Oscar Romero, and Jeffrey P Bigham. 2017. InstructableCrowd: Creating IF-THEN Rules for Smartphones via Conversations with the Crowd. In *submission to ACM Transactions on Computer-Human Interaction (TOCHI)*. ACM, 26 pages. [**Under Review. 1st–Round Decision: Major Revision**].
- [4] **Ting-Hao K. Huang** and Jeffrey P. Bigham. 2017. A 10-Month-Long Deployment Study of On-Demand Recruiting for Low-Latency Crowdsourcing. In *In Proceedings of The fifth AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2017)*. AAAI, AAAI.
- [5] **Ting-Hao K. Huang**, Joseph Chee Chang, and Jeffrey P Bigham. 2018. Evorus: A Crowd-powered Conversational Assistant Built to Automate Itself Over Time. In *Proceedings of the 2018 Conference on Human Factors in Computing Systems (CHI 2018)*. ACM, 10 pages. [**Final Review Scores: 4, 5, 4.5, 4**].

- [6] **Ting-Hao K. Huang**, Joseph Chee Chang, Saiganesh Swaminathan, and Jeffrey Bigham. 2017a. Evorus: A Crowd-powered Conversational Assistant That Automates Itself Over Time. In *Poster track of the 20th ACM Symposium on User Interface Software and Technology (UIST Poster 2017)*.
- [7] **Ting-Hao K. Huang**, Yun-Nung Chen, and Jeffrey P. Bigham. 2017b. Real-time On-Demand Crowd-powered Entity Extraction. In *Proceedings of the 5th Edition Of The Collective Intelligence Conference (CI 2017)*. New York University, NY, USA. Oral presentation.
- [8] **Ting-Hao K. Huang**, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, and others. 2016. Visual Storytelling. In *Proc. the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2016)*. NAACL.
- [9] **Ting-Hao K. Huang**, Walter S. Lasecki, Amos Azaria, and Jeffrey P. Bigham. 2016. “Is there anything else I can help you with?”: Challenges in Deploying an On-Demand Crowd-Powered Conversational Agent. In *Proceedings of AAAI Conference on Human Computation and Crowdsourcing 2016 (HCOMP 2016)*. AAAI.
- [10] **Ting-Hao K. Huang**, Walter S. Lasecki, and Jeffrey P. Bigham. 2015. Guardian: A Crowd-Powered Spoken Dialog System for Web APIs. In *Proceedings of the Third AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2015, November 8-11, 2015, San Diego, California*. 62–71.
- [11] Saiganesh Swaminathan, Raymond Fok, Fanglin Chen, **Ting-Hao K. Huang**, Irene Lin, Rohan Jadvani, Walter Lasecki, and Jeffrey Bigham. 2017. WearMail: On-the-Go Access to Information in Your Email with a Privacy-Preserving Human Computation Workflow. In *30th ACM Symposium on User Interface Software and Technology (UIST 2017)*.
- [12] Shih-Ming Wang, Chun-Hui Li, Yu-Chun Lo, **Ting-Hao K. Huang**, and Lun-Wei Ku. 2016. Sensing Emotions in Text Messages: An Application and Deployment Study of EmotionPush. In *In the Demo track of the 26th International Conference on Computational Linguistics (COLING Demo 2016)*.