Why is My Social Robot so Slow? How a Conversational Listener can Revolutionize Turn-Taking

Matthew P. Aylett, Andrea Carmantini and David A. Braude

***Abstract*— Current machine dialog systems are predominantly implemented using a sequential, utterance based, two-party, speak-wait/speak-wait approach. human-human dialog is 1) not sequential, with overlap, interruption and back channels; 2) processes utterances before they are complete and 3) are often multi-party. The current approach is stifling innovation in social robots were long delays (often several seconds) is the current norm for dialog response time, leading to stilted and unnatural dialog flow. In this paper, by referencing a light weight word spotting speech recognition system - Chatty SDK, we present a practical engineering strategy for developing what we term a *conversational listener* that would allow systems to mimic natural human turn-taking in dialogue.**

I. Introduction

> "As conversational systems (in various forms) are becoming ubiquitous, it is clear that turn-taking is still not handled very well in those systems. They often tend to interrupt the user or have very long response delays, there is little timely feedback, and the flow of the conversation feels stilted."
> Skantze [1] p1.

There has been a tendency for researchers in robotics to see language processing as beyond their domain. A traditional research model model has been to wait for language technology to be available and implement it within a robot context. However, one of the major problems in NLP and robots is that it is not examined in an applied setting where effective interaction is often more important that how clever the language processing might be. The result is that language interaction with robots has fallen very much short of physical and spatial interaction. If we want robots to use language we have to study the use of language by robots. On this basis we argue that the ideas and work presented here is of key interest to practitioners and researchers working with social robots.

Most so-called, conversational systems used by social robots are, in reality, two-party, speak-wait/speak-wait systems[1]. Human conversation in contrast, is often multi-party, allows for fluid interruption and back channeling[2]. About 10% of the speech material is overlapped, with speakers often speaking, briefly, at the same time [3]. Furthermore, human participants typically respond within 200ms [4], whereas current digital systems can spend several seconds processing before saying anything. This results in less fluid interaction and impacts the functionality of social robots in areas such as education - e.g. *"Certain occurrences of social referencing appeared exclusively in the interaction with the robot, such as an involvement of the caregiver after a delay in the dialogue occurred and the robot required too much time to provide an adequate utterance."* [5], support for older users - e.g. *"the turn-taking delays in the dialogue were significant, which hinders the communication."* [6] and results in systems being regarded as inferior and poor at carrying out their tasks - e.g. *"users not only rate the incremental system as more responsive, but also rate its recommendation performance as higher."* [7].

There is a body of previous work looking at incremental dialog processing[3] (e.g. [8], [9], [10], [11], [12], [13]). Over 20 years ago Allen et al [8] pointed out that speak-wait/speak-wait processing can *"…make the interaction unnatural and stilted, and will ultimately interfere with the user's ability to focus on the problem itself rather than on making the interaction work."*

There are toolkits available to implement incremental processing of dialog for example InproTK [14], Incremental RASA [15], Retico [16]. These systems follow a waterfall design pattern where each module can form hypotheses based on incremental input but allow replanning if these hypotheses are rejected as new data is processed. This approach has a number of severe drawbacks:

1) Processing data without end pointing and a right context often produces inferior results to a system that *waits* for an utterance to conclude.
2) The architecture is complex and difficult to debug and test.
3) The extra processing power required is multiplied across all levels of the system.

Thus, despite this work, we are unaware of any commercial social robot that makes use of incremental processing to implement human style dialog turn-taking.

In this paper we suggest a hybrid approach to the problem. Rather than retooling the entire architecture of a system to deal with incremental processing we suggest adding a new module, a *conversational listener* to the system which would allow a more flexible approach to implementing human

CereProc Ltd. Edinburgh, UK.
Corresponding author: Matthew Aylett matthewaylett@gmail.com

[1] Often a two turn question/answer architecture in many circumstances.

[2] back channeling is a term for where conversational participants react with short phrases while another is talking like 'aha', 'right', 'yeah', 'hmm' to show they are listening and understand [2]

[3] All dialog processing is incremental in some respects because you don't know what the next utterance will be. However, incremental in this context means processing before you discover the end-point of a dialog partner's current utterance.

like dialog processing. This approach is informed by four observations:

1) Systems often have a strong expectation of the type of response a user is likely to make in a dialog context.
2) Human's typically respond very quickly to dialog turns that require simple responses or contain predictable content. Whereas longer inter-turn intervals are typical when a response requires significant processing.
3) Human's often start speaking before they have decided what they are going to say.
4) Before large scale, open domain, multi-speaker, continuous speech recognition was available legacy system made good use of processor efficient key word spotting.

The work presented here is novel in that it repurposes technology that already exists with a different vision and objective. The lack of progress in incremental processing of dialog within social robotics demonstrates that this work is timely and required. In our view the issues addressed by a hybrid approach to incremental dialog processing are key for progress in social robotics.

II. The Conversational Listener

The purpose of a conversational listener is to track speakers in incoming audio in order to establish who is speaking, if a speaker is about to finish speaking, what they have been talking about in general terms, and how their emotional state might be changing over time.

The conversational listener uses automatic speech recognition (ASR) technology to support current systems by offering on-device realtime streamed information that can help dialog planning and turn-taking. The key is to allow dialog systems to carry out the most complicated processing while they are already speaking. We term this a hybrid incremental (HI) approach. This approach has already been proposed by Lala et al [17]. Here, in order to facilitate the generation of fillers to grab the floor and back channels to support the dialog, the system uses a fast incremental prosodic analysis system to suggest these actions while depending on a conventional speak-wait/speak-wait ASR system to plan dialog actions. With a conversational listener we propose extending this incremental approach to include key word spotting and to allow the key words expected to be set rapidly on a turn-by-turn basis as required as well as some metric of the emotional state of the speaker.

CereProc have been working closely with Honda Research Institute over the last 4 years with the Haru Project. The conversational listener was designed and built by the authors following observations that language interaction with Haru was slow and sub-optimal. The conversational listener is implemented both as an SDK (which can be integrated into other 3rd party systems), and as a full conversational listener to support Haru, the social robot designed and built by The Honda Research Institute. The implemented system does not currently support multiple speakers. To test the system we implemented a script follower, which allows Haru to act out a script with a human actor.

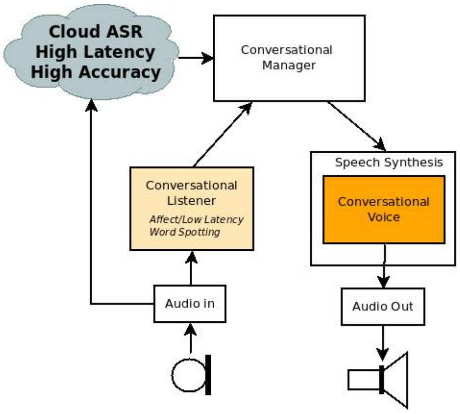

Fig. 1. Diagram showing context of the conversational listener, conversational manager and conversational TTS within the Haru system.

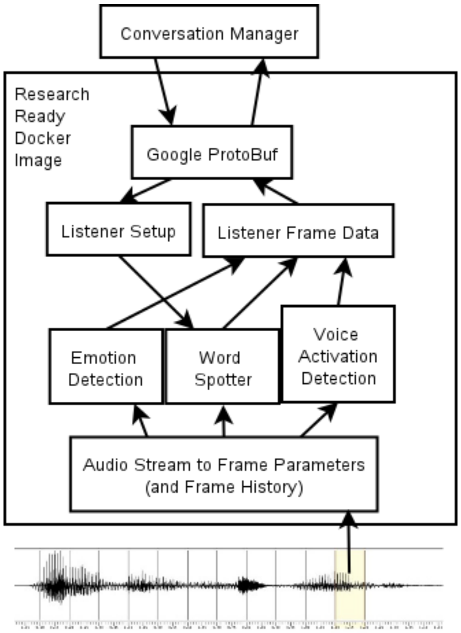

Fig. 2. Architecture of the Honda Research Institute Conversational Listener.

III. Chatty SDK implementation

Key word spotting has fallen out of favor in the ASR community, replaced either by open vocabulary multi-speaker continuous speech recognition or very rapid on device wake word spotting. However, in the late 80s and early 90s many effective systems were built based on key word spotting (e.g. [18]).

Chatty SDK is built as an underlying phone recognizer trained on a large corpus of speech data (Libraspeech [19]). A front-end processes audio input from microphone using fast voice activity detection system and feature generation. The front-end serves chunks of frames to the model to allow online inference. The ASR model is a neural network composed of unidirectional recurrent layers trained with the Connectionist Temporal Classification loss (CTC) [20]. This results in a Markovian model which runs online with a small

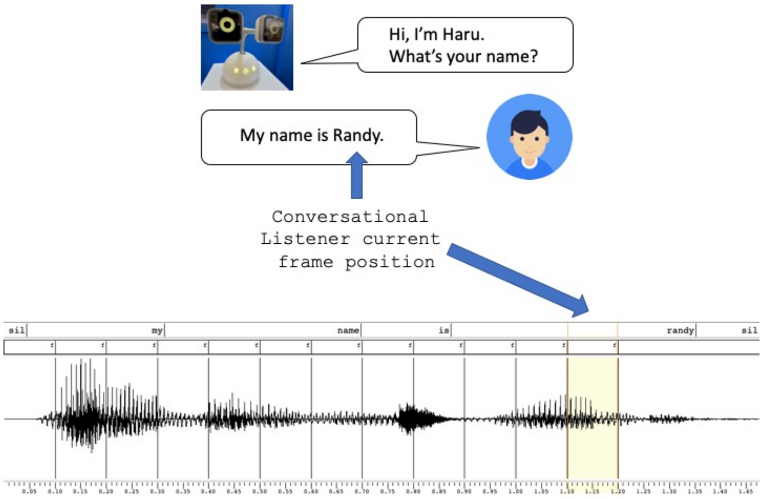

Fig. 3. Example of how a conversational listener could support an incremental dialog system.

footprint and low computing costs. Further optimization is obtained by weight quantization and phone synchronous decoding [21]. Following Hwang et al. [22] and Zhuang et al. [23], key words are searched on the phone lattice generated by the CTC model. The confidence score for each key word is determined by the posteriors output by the ASR model and the minimum edit distance with the key word phone string. This allows detection even in cases where the ASR hypothesis has errors. Key Words are fast to set, and can be changed as rapidly as utterance by utterance.

IV. Honda Research Institute conversational listener

The Chatty SDK was used as a basis to build a conversational listener as part of the Honda Research Institute Haru social robot project [24]. The conversational listener is a module which returns information on the dialog partner's speech using a 20ms frame rate in real-time. The information it returns includes key word start and end frame times, voice activation score and, for the Haru project, experimental scores aiming to detect emotion in the conversational partners voice.

The conversational listener does not aim to replace full ASR turn-by-turn processing. Rather the aim is to allow the system to make a faster decision on how to respond, ideally before the conversational partner has finished speaking (see Figure 1 and 2).

A. *Example of the conversational listener in action*

The example in figure 3 shows the beginning of a conversation between Haru and the user Randy. Haru has been woken up and asks the user's name. The user will say "My name is Randy" but has not finished saying his name with the conversational listener positioned at "My name is Ran..."

With current systems, Haru could do nothing until the end of utterance was detected. It would then send the audio to a high resource accurate cloud recognizer and when that was processed receive the recognized text. It would then process the text to form the next dialog move. All of this could create a latency of over a second.

Using the conversational listener, based on the context of the dialog (Haru has just asked the user name) and the key words *my* and *name*, Haru can hypothesize that the user has answered the question. Haru doesn't know the users name yet but can start processing a response immediately such as. "Great to get to know you" and be able to output the response within 100ms of the speaker finishing the utterance. While Haru is saying this phrase the cloud based ASR can get the correct name and plan Haru's next utterance. "How can I help Randy?" producing a completely fluid response and then await the user's instructions.

V. Conclusion

Mimicking human behavior may not matter for many applications. For example, Siri, Google Assist, and Alexa function adequately without human-human style turn taking. However, not being able to use effective elements of human behavior that are appropriate in an engineering design is severely limiting. An HI approach offers a solution where rapid linguistic information is collected to support rapid turn taking (even overlap and back channel) with a traditional speak-wait/speak-wait approach to support long term planning and complex dialog processing. This presents a challenge to the research community in terms of designing dialog managers that can deal with parallel and possibly conflicting ASR information as well as setting key words in advance to leverage prior knowledge of dialog context. Future work will focus on: 1) evaluating the Chatty SDK in a set of dialog test harnesses; 2) for Honda Research Institute to build an effective conversation manager for Haru to make use of the dynamic incremental speech information provided.

REFERENCES

[1] G. Skantze, "Turn-taking in conversational systems and human-robot interaction: a review," *Computer Speech & Language*, vol. 67, p. 101178, 2021.

[2] H. H. Clark, *Using language*. Cambridge university press, 1996.

[3] Ö. Çetin and E. Shriberg, "Analysis of overlaps in meetings by dialog factors, hot spots, speakers, and collection site: Insights for automatic speech recognition," in *ICSLP*, 2006.

[4] M. Bull and M. Aylett, "An analysis of the timing of turn-taking in a corpus of goal-oriented dialogue," in *ICSLP*, 1998.

[5] N. F. Tolksdorf, C. E. Crawshaw, and K. J. Rohlfing, "Comparing the effects of a different social partner (social robot vs. human) on children's social referencing in interaction," in *Frontiers in Education*, vol. 5. Frontiers Media SA, 2021, p. 569615.

[6] J. Oliveira, G. S. Martins, A. Jegundo, C. Dantas, C. Wings, L. Santos, J. Dias, and F. Perdigão, "Speaking robots: The challenges of acceptance by the ageing society," in *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2017, pp. 1285–1290.

[7] V. Tsai, T. Baumann, F. Pecune, and J. Cassell, "Faster responses are better responses: Introducing incrementality into sociable virtual personal assistants," in *9th International Workshop on Spoken Dialogue System Technology*. Springer, 2019, pp. 111–118.

[8] J. Allen, G. Ferguson, and A. Stent, "An architecture for more realistic conversational systems," in *Proceedings of the 6th international conference on Intelligent user interfaces*, 2001, pp. 1–8.

[9] D. Schlangen and G. Skantze, "A general, abstract model of incremental dialogue processing," *Dialogue & Discourse*, vol. 2, no. 1, pp. 83–111, 2011.

[10] H. Hastie, O. Lemon, and N. Dethlefs, "Incremental spoken dialogue systems: Tools and data," in *NAACL-HLT Workshop on Future directions and needs in the Spoken Dialog Community: Tools and Data (SDCTD 2012)*, 2012, pp. 15–16.

[11] L. Zilka and F. Jurcicek, "Incremental LSTM-based dialog state tracker," in *ASRU*, 2015, pp. 757–762.

[12] G. Skantze, "Towards a general, continuous model of turn-taking in spoken dialogue using LSTM recurrent neural networks," in *SIGDIAL*, 2017.

[13] M. Roddy, G. Skantze, and N. Harte, "Multimodal continuous turn-taking prediction using multiscale RNNs," in *ICMI*, 2018, pp. 186–190.

[14] T. Baumann, O. Buß, and D. Schlangen, "Inprotk in action: Open-source software for building german-speaking incremental spoken dialogue systems," 2010.

[15] A. Rafla and C. Kennington, "Incrementalizing rasa's open-source natural language understanding pipeline," *arXiv preprint arXiv:1907.05403*, 2019.

[16] T. Michael, "Retico: An incremental framework for spoken dialogue systems," in *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2020, pp. 49–52.

[17] D. Lala, K. Inoue, and T. Kawahara, "Smooth turn-taking by a robot using an online continuous model to generate turn-taking cues," in *ICMI*, 2019, pp. 226–234.

[18] Y. Takebayashi, H. Tsuboi, H. Kanazawa, Y. Sadamoto, H. Hashimoto, and H. Shinchi, "A real-time speech dialogue system using spontaneous speech understanding," *IEICE TRANSACTIONS on Information and Systems*, vol. 76, no. 1, pp. 112–120, 1993.

[19] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *ICASSP*, 2015, pp. 5206–5210.

[20] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[21] Z. Chen, W. Deng, T. Xu, and K. Yu, "Phone synchronous decoding with CTC lattice." in *INTERSPEECH*, 2016, pp. 1923–1927.

[22] K. Hwang, M. Lee, and W. Sung, "Online keyword spotting with a character-level recurrent neural network," *arXiv preprint arXiv:1512.08903*, 2015.

[23] Y. Zhuang, X. Chang, Y. Qian, and K. Yu, "Unrestricted vocabulary keyword spotting using LSTM-CTC." in *INTERSPEECH*, 2016, pp. 938–942.

[24] R. Gomez, D. Szapiro, K. Galindo, and K. Nakamura, ""Haru": Hardware design of an experimental tabletop robot assistant," in *2018 13th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2018, pp. 233–240.