Table 1: Case study. W2W correctly identifies addressees from preceding and subsequent utterances in a complex scenario.

Speaker	Utterance	Addressee	W2W	SIRNN
a^1	You can mount file systems without uuid too. Nah, didn't work http://paste.ubuntu.com.	a^2	a^2	a^2
a^2	Have you tried converting the source via ffmpeg? Then you type yes and press enter.	a^4	a^4	a^4
a^3	So, drivers are good? Glxgears shows 60fps all the time!	a^5	a^5	a^2
a^4	Yes with vlc still no audio dafitykins please check http://imgur.com/6phzzln.	a^2	a^2	a^2
a^5	Glxgears isn't really a good test so, do i have a bootable, usable install image on my usb or not?	a^3	a^3	a^2

1 Explanations for Why Our Model Works Better

Here we give an explanation on why W2W works better based on the case in Table 1 (*in our paper this table is Table 7 in appendix, we move it here for easy reading*).

Firstly, the PAM module facilitates more information fusion across user representations. Taking the 4-th state tracking step as an example, the utterance of this turn is a reply to the question raised by a2 at the second turn. However, the relevance between this utterance and the third utterance is small. The PAM module is designed to capture this phenomenon on similarity difference by calculating a series of weights on each listener and update the representations of the speaker and each listener correspondingly. In comparison, the baselines update the representation of each listener with the same input representation and does not incorporate listener information into the state tracking procedure of the speaker representation. The experiments from the ablation study also prove the effectiveness of the PAM module.

Secondly, the forward-and-backward scanning scheme could also benefit the performance in comparison with the forward only scheme in the baselines. We focus on the users a1 and a5 in Table 1 for comparison. In the state tracking procedure of the baselines, the representation of a1 is updated according to what he/she says at the first step. Then, the representation of a1 is updated as a listener in the following steps. While being updated to the role of listener, the model is aware of what he/she has said in the first step, which means in each step of the state tracking process, his conversational information is fully utilized. As for a5, his conversation information is only utilized in the last step and the model regards a5 and a4 to have the same representation until step 4. Therefore, imbalance exists on users of different positions and the user appears earlier in the session tends to be modeled more sufficiently. However, the forward-and-backward scanning scheme can overcome this limitation in the baselines and thus raise the performance. This can also explain why the baselines tend to make mistakes at last few turns.

Third, the baselines ignore prior information in multi-party conversations, e.g., the speakers are more likely to address to the preceding speakers in general. W2W learns the prior information by position embedding and utilizes it for a better prediction.

2 Calculation Formula of Overlapping Rate

$$
\begin{aligned}
r &= \frac{\sum_i \frac{\sum_j w_i^j \delta_i^j}{n_{u,i} * \theta_i}}{n_s} \\
\theta_i &= \sum_j w_i^j
\end{aligned}
\tag{1}
$$

The proposed Overlapping Rate r measures the consistency between the model prediction and the manual annotation. In Equation 1, r denotes the overlapping rate metric, n_s denotes the number of sessions, and $n_{u,i}$ denotes the number of effective utterances in the i^{th} session. w_i^j denotes the weight of the j^{th} utterance in the i^{th} session. In our scenario, the weight of an utterance where three annotators give the same annotation is 1.5 and the weight of an utterance where two of the three annotators give the same annotation is set to 1. θ_i is a normalization term. δ_i^j is an indicator variable, which equals to 1 when the model prediction and the human annotation is consistent on an utterance, and equals to 0 otherwise.

We will add this to the paper.