When can I Speak? Predicting initiation points for spoken dialogue agents

Siyan Li **Ashwin Paranjape** **Christopher D. Manning**
Stanford University
{siyanli, ashwinp, manning}@cs.stanford.edu

Abstract

Current spoken dialogue systems initiate their turns after a long period of silence (700-1000ms), which leads to little real-time feedback, sluggish responses, and an overall stilted conversational flow. Humans typically respond within 200ms and successfully predicting initiation points in advance would allow spoken dialogue agents to do the same. In this work, we predict the lead-time to initiation using prosodic features from a pre-trained speech representation model (wav2vec 1.0) operating on user audio and word features from a pre-trained language model (GPT-2) operating on incremental transcriptions. To evaluate errors, we propose two metrics w.r.t. predicted and true lead times. We train and evaluate the models on the Switchboard Corpus and find that our method outperforms features from prior work on both metrics and vastly outperforms the common approach of waiting for 700ms of silence.

1 Introduction

Spoken dialogue agents have exploded in popular use (e.g., Alexa, Siri, and Google Home). However, they only support explicit turn-taking mechanisms: they detect user initiation and barge-ins using wake-words and identify end of user turns based on a silence period (typically between 700–1000ms). Turn-taking feels unnatural under such mechanisms, leading to less "conversational" interactions (Woodruff and Aoki, 2003). This is particularly damaging for open-ended social conversations where thoughtful silences get wrongly interrupted (Chi et al., 2021). To fix this issue, we predict initiation opportunities for spoken dialogue agents for both turn-taking and backchanneling.

Prior work predicting initiation points uses prosodic features like pitch and frequency variation with bag-of-embeddings to predict backchannels (Ruede et al., 2017a) and turn-completion (Skantze, 2017), and more recently, Ekstedt and Skantze (2021) finetuned GPT-2 on dialogue datasets to predict turn-completion using only word features. However, they either predict a binary label indicating initiation in a wide event horizon, which is imprecise; or they predict a binary label for an initiation to happen at a set offset in the future, in which case a single incorrect prediction leads to a missed initiation.

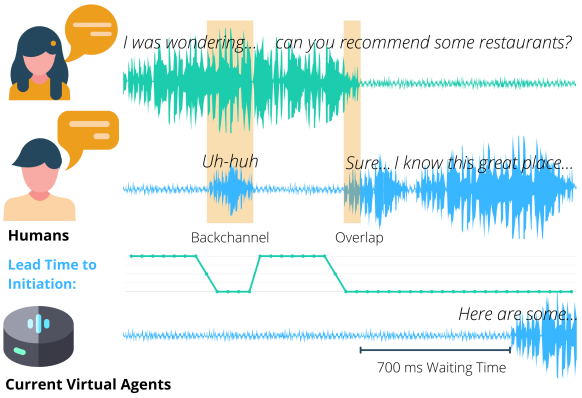

Figure 1: Humans produce overlapping speech with small gaps. By predicting lead to initiation, virtual agents can respond without long waiting periods

As a robust generalization of previous approaches, we predict the lead time to initiation as a continuous value. We model initiation (next utterance from a different speaker) directly and not end-of-turn because there is a variable (and possibly negative) gap between the two (Skantze, 2021). In this work, we combine two models: wav2vec 1.0 (Schneider et al., 2019) for representing prosodic features and finetuned GPT-2 (Radford et al., 2019) for word features. We model the task with a Gaussian Mixture Model (GMM) to account for inherent uncertainty. We train and evaluate our models on Switchboard (Godfrey et al., 1992) and find that the combination of the pretrained models performs the best, vastly outperforming a silence-based baseline that waits for 700ms of silence and baselines using features from prior work.

Proceedings of the SIGdial 2022 Conference, pages 217–224
Heriot-Watt University, Edinburgh, UK. 07-09, September, 2022

2 Related Work

Prior work for dialogue turn-taking either uses silent gaps as cues or predicts future events repeatedly. A key issue with systems that use silent gaps as initiation cues (Huang et al., 2011; Cohen et al., 2004; Witt, 2015) is the difficulty of adjusting the silence thresholds to accommodate dialogue states (Skantze, 2021). When predicting turn-taking repeatedly, i.e. predicting future actions at every timestep, acoustic features such as pitch and frequency are often used, with additional linguistic features including part-of-speech or word embeddings (Ruede et al., 2017a,b; Skantze, 2017; Ward et al., 2018; Roddy et al., 2018). More recently, Ekstedt and Skantze (2021) implement a spoken dialogue system for travel conversations using TurnGPT (Ekstedt and Skantze, 2020). However, a short silence threshold is still used to determine initiation of agent responses.

Outside of dialogue, Neumann et al. (2019) propose probabilistic models for predicting events in videos, Lei et al. (2020) forecast frames and Vondrick et al. (2016) forecast actions. Time-to-event analysis in the medical domain involves modeling patient status as a function of time (Meira-Machado et al., 2009; Soleimani et al., 2017).

3 Methods

3.1 Setup

$\mathbf{I}^{\mathbf{k}}_{\text{spkr}}$ is the time of k-th initiation (both backchannels and transitions) by a speaker. We use the **current speaker**'s audio and transcript information to predict the the **lead time to initiation**, $\hat{\tau}_t$, of the **target speaker**. When the current speaker is speaking, we consider an **event horizon** δ_{max} to narrow the prediction range and at time t, define the true **lead time to initiation** as $\tau_t = \min(\delta_{max}, I^k_{\text{tgr}} - t)$. When the target speaker is speaking, we set $\tau_t = 0$, to ensure a well-balanced distribution.

3.2 Models

We make two novel contributions. First, we fuse rich contextual prosodic features from a pretrained wav2vec model with contextual word representations from a pretrained GPT-2 model. Prior work has not used such rich contextual prosodic features nor their combination with word representations. Second, prior work does not model the inherent uncertainty of initiations. Inspired by the video event prediction literature (Neumann et al., 2019), we do this using a Gaussian mixture model and maximize model likelihood under the data distribution.

3.2.1 Features

Features are extracted from the current speaker's voice channel and transcript. We suffix model names with abbreviated versions of the features they use.

Wav2vec Embeddings (W): Raw audio is fed into Wav2vec 1.0 (Schneider et al., 2019) to obtain convolutional embeddings. We choose Wav2vec 1.0 because of its unidirectional nature, which enables handling efficient incremental processing of audio. We keep the model weights frozen.

GPT-2 Embeddings (G): This is the GPT-2 Small (Radford et al., 2019) embedding of the last salient word from the target speaker after feeding in prior utterances. The embedding is updated incrementally as more utterances are transcribed. We fine-tune the GPT-2 model during training.

RMSE (R): We select the Root Mean Square Energy (RMSE) of the raw waveform to signal current speaker silence. It simulates audio energy and power in features from prior work.

Additional Prosodic Features (A): Previous work explores pre-neural prosodic features (Ruede et al., 2017a,b; Skantze, 2017); to compare our approach with previous approaches, we include pitch and frequency, both represented as a number for each frame. The prosodic features, including RMSE, are calculated with a frame shift of 50 ms and a window length of 100 ms. Additional details for feature implementation are in Appendix A.1.

Wav2vec features are subsampled to 50 ms by selecting embeddings at every 50ms and for other audio features by adjusting the frame shift. Audio features are concatenated and input to an LSTM network. When GPT-2 embeddings are used, they are concatenated with the LSTM's final hidden state. This is fed into a linear head. More training details are presented in Appendix A.2.

3.2.2 Gaussian Mixture Model

There is an inherent uncertainty in the precise location of an initiation (e.g., it can occur a few milliseconds before or after the prediction) and a single Gaussian is sufficiently powerful to model it because the uncertainty is localized. However, a speaker can initiate at many points in time that are far apart, for e.g., at the completions of grammatical clauses that can happen hundreds of milliseconds apart. We use a Gaussian mixture

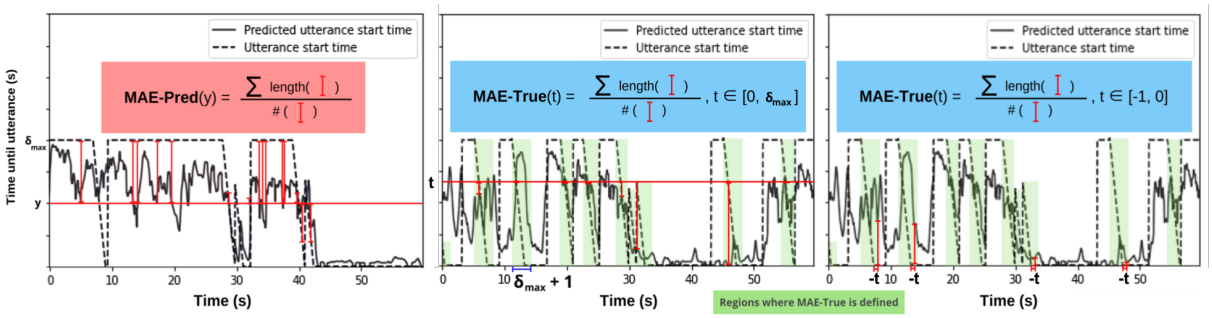

Figure 2: An explanation of our metrics. The red vertical intervals correspond to $|\tau_x - \hat{\tau}_x|$ in the equations. As illustrated, MAE-Pred(y) evaluates the expected error when a model predicts value y. For MAE-True(t), we highlight the regions where MAE-True can be calculated in green; depending on how long the current speaker's next utterance is, the region has a maximum length of $\delta_{max} + 1$.

model (GMM) to capture this multimodal prediction space.

At every time step, we predict the parameters: mean, variance and weights, for T Gaussian distributions $\{\mu, \sigma, h\}_{[1..T]}$. The training objective is to maximize the log of the summed likelihood of τ_t:

$$\log\Big(\sum_{i=1}^{T} h_i \cdot \frac{1}{\sigma_i\sqrt{2\pi}} \cdot \exp - \frac{(\tau_t - \mu_i)^2}{2\sigma_i^2}\Big)$$

At inference, we use the mean of the Gaussians.

3.2.3 Baselines

Silence Baseline: We compare our models with an RMSE-based non-neural baseline. We detect voice activity based on whether RMSE is above a certain threshold (0.01 for this work). If there is a gap of more than 700ms in voice-activity, the baseline predicts an initiation $\tau_t = 0$ at the current time, otherwise predicts δ_{max}.

GMM-AG: We use this baseline as a proxy for Ruede et al. (2017a), where pitch, power, and FFV are used as the prosodic features, and word2vec embedding of the most recent salient word is the linguistic feature. We simulate these features using RMSE, pitch and frequency (the prosodic features), and GPT-2 embeddings.

GMM-G: Ekstedt and Skantze (2020) use GPT-2 to emulate possible continuations of the current conversation in order to decide turn-relevant places. Although we do not use the same algorithm, we still use GPT-2 embedding as a feature. We train a GMM on last-salient-word GPT-2 embeddings only, and use this as a representative baseline for Ekstedt and Skantze (2020).

GMM-WGR-1: We train a Gaussian mixture model with $T = 1$ Gaussian to examine whether using multiple Gaussian models to capture different factors for utterance timing is necessary. This model is trained on the same data as our GMM-WGR model, with Wav2vec, GPT-2, and RMS features.

3.3 Training and Evaluation Data

For training, we randomly sample 60 second audio segments that have its first target speaker initiation in the first 5 to 10 seconds. This is to make sure that there is at least one initiation with enough context. We backpropagate losses only in a limited range around each initiation I^i_{tgt}, $[I^i_{\text{tgt}} - 2\delta_{max}, I^i_{\text{tgt}} + 1]$ This is to ensure a balanced distribution of τ_t. For evaluation and testing, we instead cover entire dialogues by collecting 60-second segments every 20 seconds. We randomly choose the target speaker for each segment.

3.4 Metrics

To measure the performance of our models that produce continuous values, previous work's classification-based metrics are insufficient to differentiate between a prediction error of 0.2 versus 2 seconds. Additionally, we want to differentiate between how precise model predictions are and how well they cover the initiations observed in the dataset. We improve upon Time-to-event error from Neumann et al. (2019), and propose Mean Absolute Error w.r.t. Predicted Lead Time (MAE-Pred) and Mean Absolute Error w.r.t. True Lead Time (MAE-True) as analogues of precision and recall that improve existing metrics (Skantze, 2017). If a practitioner needs l seconds to generate a response, MAE-Pred(l) gives the expected error when the model predicts l (precision) and MAE-True(l) gives the expected error with the true lead

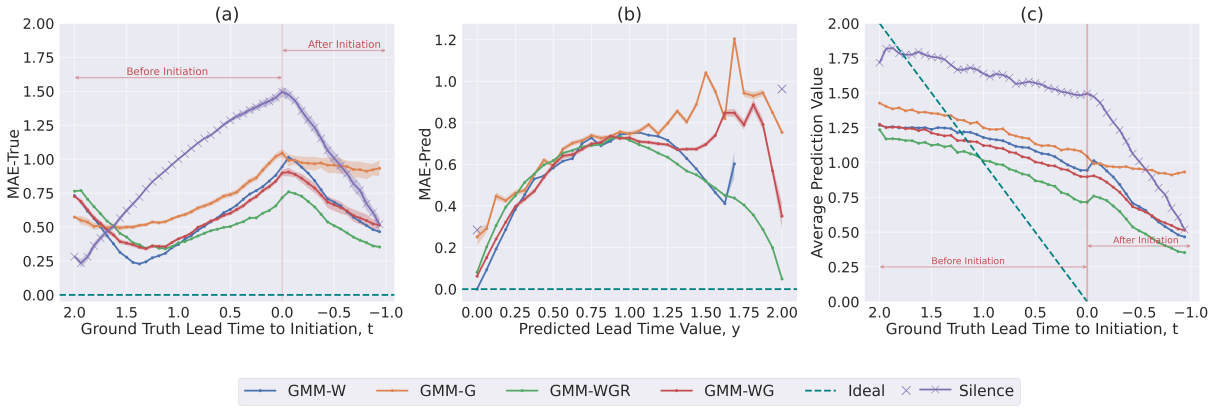

Figure 3: (a) MAE-True, (b) MAE-Pred, and (c) average predicted lead time values for representative neural models and the silence baseline. 95% C.I. are represented by the lightly shaded regions. A perfect model would achieve the "ideal" (dashed) lines. In (b), because the silence based model only predicts 0 or δ_{max}, only these two points are defined in plot (b) for the silence based baseline. The corresponding MAE-Pred values for the silence baseline are indicated as crosses in plot(b). All of our models, including the best performing GMM-WGR, significantly outperform the silence-based model that waits for 700 ms.

time is l (recall). With the set S representing the timesteps included in the calculations, both metrics can be represented as

$$\sum_{x \in S} |\tau_x - \hat{\tau}_x| / |S|$$

Specifically, for **MAE-Pred**(y):

$$S = \{x | \hat{\tau}_x = y\}, y \in [0, \delta_{max}]$$

For **MAE-True**(t):

$$S = \{I^i_{\text{tgt}} - t\}, t \in [-1, \delta_{max}] \cap [I^i_{\text{tgt}} - I^{j+1}_{\text{cur}}, I^i_{\text{tgt}} - I^j_{\text{cur}}]$$

for all target-speaker initiations I^i_{tgt}, limiting to intervals between two consecutive initiations by the current speaker. When $t \leq 0$, the initiation has already occurred and $\tau_t = 0$. We quantize both true and predicted values into 16 buckets per second.

As an aggregated metric, we propose Macro-MAE (MMAE). We define MMAE-X$(a, b) = \sum_{v \in S_{ab}}$MAE-X$(v)/|S_{ab}|$, where S_{ab} is the set of bucket values between a and b for a given set S. We define 1 second before and $0.5s$ after initiation as the interval of interest for MMAE-True, and similarly predicted values between 0 and 1 for MMAE-Pred. We compute MMAE = MMAE-True$(-0.5, 1)$ + MMAE-Pred$(0, 1)$ as a single number quantifying model performance.

4 Experiments

For training and evaluation, we use audio conversations from Switchboard (Godfrey et al., 1992). We select a random set of 200 training, 20 validation, and 20 test dialogues out of a total of 1000 dialogues due to computational constraints. We use the validation set to select the best performing checkpoint based on MMAE scores and report the numbers on the test set. For the GMM models, we experimented with $T = 1, 5, 10, 15, 20$, and found $T = 15$ to be the best-performing. [1]

We plot the MAE-Pred and MAE-True values in Figure 3 and the show the MMAE values in Table 1. A perfect model would have 0 error. As a diagnostic tool, we also plot the average prediction for each t used in MAE-True (Figure 3 (c)). Here, we expect a perfect model to be a line with a slope of -1 passing through the origin before flattening out at 0. We see that for all models MAE-True peaks (roughly) at initiation (Figure 3 (b))). Despite all the cues leading up to an initiation in the data, it is still highly optional and the models aren't able to predict it perfectly. Soon afterward, as the target speaker stays silent the models predict smaller lead times to initiation (steeper downward slope in Figure 3 (c)) and the MAE-True reduces. On the other hand, for all trained models (GMM-*), we see that MAE-Pred reduces for smaller values of y (Figure 3 (c)) indicating that the trained models are very precise when they predict near-term initiations.

Our models outperform the silence baseline by a large margin in most time windows prior to and

[1] Our code for the models and for training is available at https://github.com/siyan-sylvia-li/icarus_final

	Eval			Test
Model	MT	MP	MMAE	MMAE
GMM-AG	0.90	0.63	1.53	1.51
GMM-G	0.90	0.60	1.50	1.42
GMM-WGR-1	0.67	0.59	1.26	1.30
Silence*	1.33	0.60	1.93	1.88
GMM-W	0.70	**0.49**	1.19	1.22
GMM-WG	0.67	0.51	1.18	1.19
GMM-WGR	**0.63**	0.52	**1.15**	**1.11**

Table 1: Performance of different models on the evaluation and the test dialogues, as measured Macro-MAE values. MT = MMAE-True$(-0.5, 1)$, MP = MMAE-Pred$(0, 1)$. * Since only 0 and δ_{max} are valid predictions for Silence Baseline, we use (MAE-Pred(0) + MAE-Pred(δ_{max}))/2 as MMAE-Pred$(0, 1)$.

after initiations (Figure 3 and Table 1). GMM-WGR outperforms prior work baselines: GMM-G (TurnGPT) and GMM-AG (Ruede et al. (2017a)).

Comparing GMM-WG vs. GMM-G, Wav2vec features reduce MAE-True after initiation and stabilizes MAE-Pred for small predicted lead times; GMM-G's predictions stay constant after initiations, because it can only access the transcript from the current speaker. Comparing GMM-WG vs. GMM-W, GPT-2 features reduce MAE-True near initiations, possibly because they provide the model with word cues. GMM-WGR has a lower MMAE-True(-0.5, 1) compared to GMM-WG, indicating that Wav2vec doesn't capture silences as well as RMSE. GMM-WGR-1, our baseline with one Gaussian, performs poorly compared to GMM-WGR, highlighting the importance of the Gaussian mixture.

5 Conclusion

We present the task of lead time to initiation prediction as a continuous-valued problem, collapsing transition and backchannel timing problems into one. We additionally propose metrics to capture precision and coverage in these predictions. Our models trained on pretrained prosodic and verbal embeddings consistently outperform the commonly-used silence baseline. We believe our work will build a foundation for more naturalistic virtual agents with human-like conversational behaviors.

References

Ethan A Chi, Caleb Chiam, Trenton Chang, Swee Kiat Lim, Chetanya Rastogi, Alexander Iyabor, Yutong He, Hari Sowrirajan, Avanika Narayan, Jillian Tang, et al. 2021. Neural, neural everywhere: Controlled generation meets scaffolded, structured dialogue. *Alexa Prize Proceedings*.

Michael H Cohen, Michael Harris Cohen, James P Giangola, and Jennifer Balogh. 2004. *Voice user interface design*. Addison-Wesley Professional.

Erik Ekstedt and Gabriel Skantze. 2020. TurnGPT: a transformer-based language model for predicting turn-taking in spoken dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2981–2990, Online. Association for Computational Linguistics.

Erik Ekstedt and Gabriel Skantze. 2021. Projection of turn completion in incremental spoken dialogue systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 431–437, Singapore and Online. Association for Computational Linguistics.

J.J. Godfrey, E.C. Holliman, and J. McDaniel. 1992. Switchboard: telephone speech corpus for research and development. In *ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 517–520 vol.1.

Lixing Huang, Louis-Philippe Morency, and Jonathan Gratch. 2011. Virtual rapport 2.0. In *International workshop on intelligent virtual agents*, pages 68–79. Springer.

Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. 2020. What is more likely to happen next? video-and-language future event prediction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8769–8784, Online. Association for Computational Linguistics.

Luís Meira-Machado, Jacobo de Uña-Álvarez, Carmen Cadarso-Suárez, and Per K Andersen. 2009. Multi-state models for the analysis of time-to-event data. *Statistical Methods in Medical Research*, 18(2):195–222. PMID: 18562394.

Lukáš Neumann, Andrew Zisserman, and Andrea Vedaldi. 2019. Future event prediction: If and when. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2935–2943.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Matthew Roddy, Gabriel Skantze, and Naomi Harte. 2018. Investigating speech features for continuous turn-taking prediction using LSTMs. *arXiv preprint arXiv:1806.11461*.

Robin Ruede, Markus Müller, Sebastian Stüker, and Alex Waibel. 2017a. Enhancing backchannel prediction using word embeddings. In *Interspeech*, pages 879–883.

Robin Ruede, Markus Müller, Sebastian Stüker, and Alex Waibel. 2017b. Yeah, right, uh-huh: A deep learning backchannel predictor. *CoRR*, abs/1706.01340.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. *arXiv preprint arXiv:1904.05862*.

Gabriel Skantze. 2017. Towards a general, continuous model of turn-taking in spoken dialogue using LSTM recurrent neural networks. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 220–230, Saarbrücken, Germany. Association for Computational Linguistics.

Gabriel Skantze. 2021. Turn-taking in conversational systems and human-robot interaction: a review. *Computer Speech & Language*, 67:101178.

Hossein Soleimani, James Hensman, and Suchi Saria. 2017. Scalable joint models for reliable uncertainty-aware event prediction. *IEEE transactions on pattern analysis and machine intelligence*, 40(8):1948–1963.

Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. 2016. Anticipating visual representations from unlabeled video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 98–106.

Nigel G Ward, Diego Aguirre, Gerardo Cervantes, and Olac Fuentes. 2018. Turn-taking predictions across languages and genres using an LSTM recurrent neural network. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 831–837. IEEE.

Silke Witt. 2015. Modeling user response timings in spoken dialog systems. *International Journal of Speech Technology*, 18(2):231–243.

Allison Woodruff and Paul M Aoki. 2003. How push-to-talk makes talk less pushy. In *Proceedings of the 2003 international ACM SIGGROUP conference on Supporting group work*, pages 170–179.

A Appendix

A.1 Feature Implementation

1. Pitch: https://pytorch.org/audio/main/functional.html#compute-kaldi-pitch

2. Frequency: https://librosa.org/doc/main/generated/librosa.yin.html

3. Root Mean Square Energy: https://librosa.org/doc/main/generated/librosa.feature.rms.html

A.2 Training Details

The models are trained on one A100 GPU. All model LSTM's have two layers with 128 hidden units. Each epoch approximately last 1000 seconds, and we train each neural model for 7 epochs, at which point overfitting would have definitely occurred. We train all models with dropout 0.1, Adam optimizer, and a weight decay of 0.0001. We include a comprehensive list of our models and their training details in Table 4.

A.3 Additional Model: Heuristic Heatmap

We have tried training another probabilistic model from Neumann et al. (2019), Heuristic Heatmap. We did not find this model to significantly outperform our GMM-Full model, although it does exhibit interesting qualities.

Heuristic Heatmap (Histogram-based Density Estimator): This model captures temporal shifts in the probability distribution of lead time; as the current speaker keeps speaking, the likelihood of an imminent initiation increases for the target speaker, shifting the probability mass from higher to lower lead time values. At every time step, the model produces a probability distribution with $2\delta_{max}r$ ($r = 16$, the resolution of our estimates) bucket values $h_i = P(\tau_t = \frac{2\delta_{max}i}{2\delta_{max}r})$. Training minimizes the difference between the predicted distribution and a Gaussian centered at τ_t. During inference, the prediction bucket with the highest probability is returned.

Model	W	G	Ac	R
GMM-AG		✓	✓	✓
GMM-G		✓		
GMM-W	✓			
GMM-WG	✓	✓		
GMM-WGR	✓	✓		✓
Heatmap-WGR	✓	✓		✓
GMM-WGR-1	✓	✓		✓

Table 2: The trained models and their features. **W** represents Wav2vec features, **G** GPT-2 embeddings, **Ac** the set of acoustic features (pitch and frequency), **R** the RMSE of the current speaker waveform.

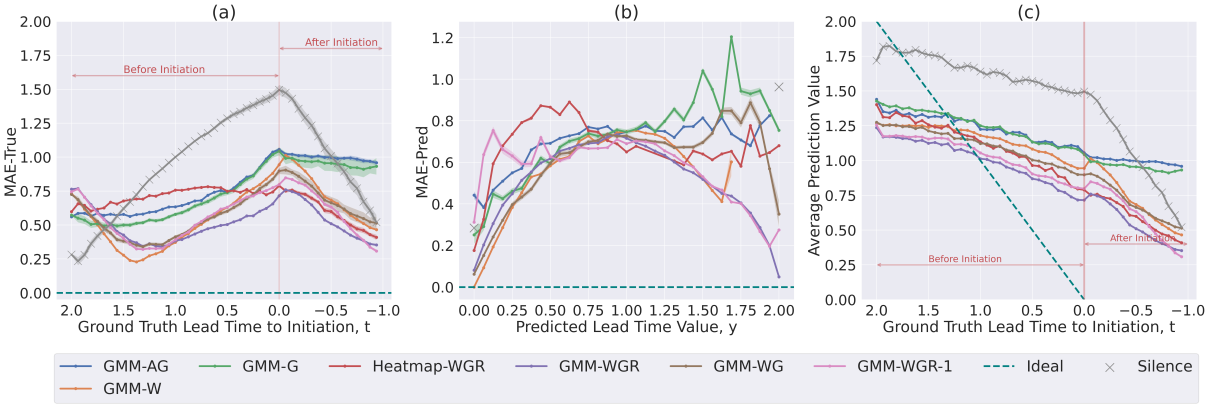

Figure 4: MAE-True, MAE-Pred graphs for all trained models. We also include the graph of average predicted lead time values given true lead time to initiation.

Model	$\mathbf{MT}_{\mathbf{Eval}}$	$\mathbf{MP}_{\mathbf{Eval}}$	$\sum_{\mathbf{Eval}}$	$\sum_{\mathbf{Test}}$
GMM-AG	0.90	0.63	1.53	1.51
GMM-G	0.84	0.58	1.50	1.42
GMM-W	0.70	0.49	1.19	1.22
GMM-WG	0.67	0.51	1.18	1.19
GMM-WGR	0.63	0.52	1.15	1.11
Heatmap-WGR	0.80	0.68	1.48	1.44
GMM-WGR-1	0.67	0.59	1.26	1.30
Silence*	1.33	0.60	1.93	1.88

Table 3: Performance of different models on the evaluation and the test dialogues, as measured by the sum of (1) the average MAE-True(t) on $t \in [1, -0.5]$ ($\mathbf{MT}_{\mathbf{Eval}}$ and $\mathbf{MT}_{\mathbf{Test}}$) and (2) the average MAE-Pred(y) on $y \in [0, 1]$ ($\mathbf{MP}_{\mathbf{Eval}}$ and $\mathbf{MP}_{\mathbf{Test}}$). * For the Silence baseline, since only 0 and δ_{max} are valid prediction values, we calculate the average of MAE-Pred(0) and MAE-Pred(δ_{max}) as $\mathbf{MP}_{\mathbf{Eval}}$ and $\mathbf{MP}_{\mathbf{Test}}$.

A.4 MAE-True and MAE-Pred on All Models

We also include the graphs for MAE-True, MAE-Pred, and average predictions per ground truth time to initiation values for all of our models. They are presented in Figure 4.

Model	Features	Learning Rate	Batch Size
GMM-AG	Acoustic features, GPT-2	1e−4	16
GMM-G	GPT-2 embedding	1e−4	16
GMM-W	Wav2vec representations	1e−4	32
GMM-WG	Wav2vec and GPT-2	1e−5	16
GMM-WGR	Wav2vec, GPT-2, and RMSE	1e−5	32
GMM-WGR-1	Wav2vec, GPT-2, and RMSE	1e−5	15
Heatmap-WGR	Wav2vec, GPT-2, and RMSE	1e−4	32

Table 4: Set of trained models.