

HAIKUS2S: A CASCADED SYSTEM FOR RESPONDING IN VERSE

Devangi Sharma^{1*}, Sophia Judicke^{1*}, Glenda Tan^{1†}, Conrad Schaumburg^{2‡}, Shinji Watanabe¹

¹ Carnegie Mellon University ² Unaffiliated

{sjjudicke, devangis, glendat, swatanab}@andrew.cmu.edu, conradschaumburg@gmail.com

ABSTRACT

Expressive speech synthesis has advanced through prosody modeling, yet generating structured poetic speech, such as haiku, remains challenging. Prior work on prosody transfer improves expressiveness, and fine-tuned poetry TTS (text-to-speech) systems capture verse intonation. However, these methods do not model haiku’s 5-7-5 syllable structure or line-ending pauses. We present a cascaded system, HaikuS2S, combining ASR (automatic speech recognition), LLM (large language model)-generated haiku, and TTS fine-tuning on both prose and custom haiku datasets. Our evaluation focuses on emotion similarity, speech quality, and prosody alignment. In our experiments, we see that our prosody and tonal alignment improve significantly with our fine-tuned systems, particularly the one trained on both general poetry and haiku. We also see that we maintain similar emotion similarity scores across all systems.

Index Terms— prosody, poetry, text-to-speech, style cloning

I. INTRODUCTION

Neural TTS and prosody modeling have improved expressive speech synthesis, but generating structured poetic speech like haiku is still difficult. Standard TTS often fails to reproduce poetic rhythm, line breaks, or emotional nuances. Prosody transfer methods enable expressive speech via learned embeddings [1], and poetry TTS systems can capture verse intonation [2]. Emotion-aware models such as emotion2vec further enhance expressiveness [3]. Yet, existing approaches ignore haiku-specific constraints like the 5-7-5 syllable structure and line-ending pauses. We address this with HaikuS2S, a cascaded system that integrates ASR, LLM-based haiku generation, and TTS fine-tuning on prose and custom haiku datasets, aiming for accurate prosody, rhythm, and expressive naturalness.

The motivation for focusing specifically on haiku generation stems from the inherent structure of haiku, which provides clear, enforceable guidelines for TTS systems. Unlike free-verse poetry, where rhythm and pauses are more abstract and subjective, haiku offers a concrete framework that can be directly modeled and learned. This makes haiku an ideal test case for exploring how well TTS can adapt to highly structured poetry. Moreover, a cascaded system gave us more fine-grained control to enforce these constraints versus an end-to-end system. The 5-7-5 format of a haiku provides an easy pattern for a TTS to identify; therefore, this work explores how varying domain specificity in training data impacts TTS performance on haiku pronunciation. This study investigates whether a generic poetry dataset is sufficient to capture haiku recitation. Our work builds

on the intersection of expressive prosody control, cascaded systems, and domain-specific fine-tuning, offering a new approach to creating expressive and accurate synthetic speech for structured poetic forms. Our contributions also include making our final system model and training and testing data publicly available via HuggingFace¹.

For HaikuS2S to succeed in haiku performance, we want it to correctly capture the prosody of a given haiku. This includes pausing on line breaks and correctly pausing, stopping, or changing tone on different punctuation marks. We also ensure our LLM outputs a valid 5-7-5 haiku.

II. RELATED STUDIES

Research on expressive text-to-speech and prosody modeling has grown rapidly in recent years.

Prosody Transfer and Control: Prosody transfer techniques learn latent prosody embeddings that enable TTS models to replicate prosodic patterns from reference audio [1]. Approaches have been proposed for fine-grained prosody control by modeling prosodic features such as duration and fundamental frequency for expressive TTS [4], enabling control of prosody at fine levels.

Emotion and Prosody Representation: Modeling emotional content in speech has become crucial for expressive TTS. Self-supervised speech representation models like emotion2vec capture emotion across multiple languages and tasks [3]. Multi-scale emotion modeling and control frameworks further enhance expressive synthesis by representing both global and local prosodic variations [5], while disentanglement approaches separate prosody and timbre for better cross-speaker emotion transfer [6].

Poetry Synthesis and Prosodic Structure: PoeticTTS demonstrates that fine-tuning on verse data yields improved poetic intonation in TTS outputs [2]. SPARSAR presents a computational poetry analysis pipeline which combines syntactic, semantic, and prosodic information for expressive poetry reading [7]. Semantics for expressive TTS explores how linguistic semantics contribute to naturalness and expressiveness in synthesized speech [8]. However, these methods do not explicitly enforce strict constraints like haiku’s 5-7-5 syllable structure and poetic pauses.

Structured Data and Low-Resource TTS: Low-resource expressive synthesis research highlights the challenges of modeling prosody and emotion with limited training data [9]. End-to-end models focusing on expressive prosody across multi-sentence contexts show that longer context and richer text features improve prosodic coherence [10].

Together, these studies highlight advances in expressive TTS, prosody modeling, and poetic synthesis. Our work builds on this literature by combining expressive prosody control, emotion representation, and domain-specific fine-tuning on haiku data for structured

*Equal first-authorship contribution

†Equal second-authorship contribution

‡Work completed at Carnegie Mellon University

¹<https://huggingface.co/HaikuS2S>

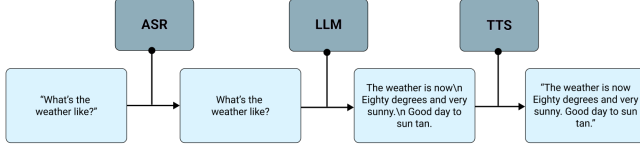


Fig. 1: Block diagram showing our cascaded haiku spoken dialog system deployed in ESPnet-SDS. Displays example user query and example haiku response.

poetic speech synthesis. Restricting to a cascaded system and focusing on haiku-specific constraints made the problem easier to tackle, as notions of prosody and success were less abstract and had more explicit rules.

III. SYSTEM DESIGN AND METHOD

HaikuS2S takes the form of a single-speaker cascaded system (see Fig. 1). We use ESPnet [11] [12] framework with `espnet/owsm_v4_base_102M` as our ASR model [13], `microsoft/Phi-3-mini-4k-instruct` [14] as our LLM, and `kan-bayashi/vctk_multi_spk_vits` as our TTS model [15]. We believe that a pretrained-only model, such as the mentioned multispeaker VITS model, provides an accurate baseline to see how a cascaded system performs at haiku recitations, as the model is trained only on novel-based and conversational texts. In our LLM prompt, we include directions to respond to the user in haiku format:

“You are a haiku poet. Reply with exactly one haiku in English: three lines with a 5-7-5 syllable pattern (five syllables, then seven, then five). No title, no preamble, no numbering—only the three lines. The haiku should reflect or respond to the user’s words.”

We have two experimental stages of HaikuS2S, which use the same ASR model, LLM, and LLM prompt. We follow two different TTS fine-tuning stages:

1. **Poetry fine-tuning** on the `mythincinifinity/libritts_r` [16] dataset of standard poetry readings, improving general prosody, intonation, and naturalness for spoken poetry.
2. **Haiku fine-tuning** on our custom haiku dataset, enforcing haiku-specific features such as 5-7-5 syllable structure and line-ending pauses.

Our proposed methods solve our task of responding to user queries in spoken haiku format by aiding in the prosody of haiku responses. Stage 2 is unique and distinct from prior results because, to our knowledge, nobody has performed hyperspecific fine-tuning on haiku data for the purpose of haiku reading correctness and tonality, and prior work has not combined strict poetic form enforcement with prosody-aware TTS fine-tuning in a cascaded generation pipeline. This two-stage fine-tuning will introduce prosodic control in stage 1, allowing for better convergence at stage 2 despite a very small stage 2 training set.

HaikuS2S ensures correct 5-7-5 haiku formatting via G2P (grapheme-to-phoneme) validation, which counts the number of vowels in each line, after a haiku is generated. This occurs between the LLM and TTS steps, where we prompt the LLM to regenerate the haiku until it satisfies the 5-7-5 format. We also introduce a special token `[\n]` in stages 1 and 2 (but not the baseline) to indicate new lines, which we hypothesize will implicitly enable better prosodic

control. This token should allow the model to learn to pause for the appropriate length following each line, mimicking natural recitation. The token was left as a unique token in the TTS dictionary and served as an indication of line pauses in the haiku audio files, allowing for longer pauses compared to the standard newline token. This system is deployed with Gradio on a T4 GPU [17].

IV. EXPERIMENTS

IV.1. Datasets

Training Data: Our first round of training used 827 poetry samples from the HuggingFace dataset `mythincinifinity/libritts_r`, which we selected via dataset curation. For our second round of training, we created a dataset of 500 haiku by having an LLM generate 500 haiku of varying topics. We then had two members of our team each record 250 distinct haiku in quiet settings using cell phones at 16kHz.

Test Data: We created a dataset of 498 haiku by having an LLM generate 500 haiku about varying topics (of which 2 were later removed for not following a 5-7-5 haiku structure). We then had two members of our team each record 250 of the haiku in a quiet setting using cell phones at 16kHz. These team members were different from those who had recorded the training data.

IV.2. Metrics

For all our objective evaluations, we used the VERSA toolkit [18] [19]. The accuracy and intelligibility metrics we used are WER (word-error rate) and CER (character error rate). These metrics were scored by performing ASR (model: `owsm_large-v3`) on our TTS outputs.

The tone and prosody metrics we used are the following:

emo_similarity: This metric compares the emotions of the reference speech to the emotions of the output audio within a range of $(-1, 1)$, where 1 is more similar and -1 is less similar.

nisqa: An estimate of conversational speech quality and naturalness within a range of $(1, 5)$, where 1 is poor, and 5 is excellent. From this metric set, we focused on `mos`, `noi`, and `col` metrics, because these indicate the overall (1) naturalness, (2) noisiness, and (3) coloration of the output compared to the reference audio.

chroma_alignment: This metric compares the alignment of pitch, palette, and harmonic content of input text and output audio within non-negative numbers, where 0 is more similar and higher numbers are less similar. From this metric set, we focused on three unscaled `cosine_dtw` metrics to indicate our normalization of pitch and consistency of melody in cadence for poetry readings.

Human Evaluation: This metric indicates the human-perceived naturalness, speed, tone, cadence, rhythm, and enjoyment, and likelihood of reuse (`lor`) for each system. We gathered this data asynchronously across 35 subjects on a Likert (1–5) scale. This study was IRB-approved and utilized general listeners with no specialty in haiku or poetry.

These metrics are appropriate because they compare the emotional and tonal aspects one would expect when reciting haiku, as well as the timing of the speech itself. By assessing the above characteristics of output speech, we believe that we are measuring the most important aspects of poetry recitation.

IV.3. Experimental Setups

We implement a two-stage fine-tuning strategy for the TTS model, defined in Section 3.

Training details: For both stages, we train the TTS model with a batch size of 16 using the Adam optimizer. For stage 1, the TTS model was trained for 150 epochs with a learning rate of $1e-4$; for stage 2, for 600 epochs and a learning rate of $5e-4$.

Testing details: For the baseline and both stages, we evaluate on the test set defined in Section 4.1.

IV.4. Results and Discussion

Speaker Intelligibility: Stage 2 had the lowest error rate across all three systems. Stage 2 achieved the lowest error rates (Fig. 2), with a more pronounced improvement in WER than CER. We hypothesize that this is because rhythm is taken into account by `owsm_wer` and `owsm_cer`. Since our baseline and stage 1 systems do not identify pauses with the same level of correctness as stage 2 of HaikuS2S, it is expected that they are considered less intelligible. **Emotion Similarity:** All systems maintained a calm emotional tone, though Stage 2 yielded statistically significant improvements in matching the exact target emotional similarity (Fig. 3).

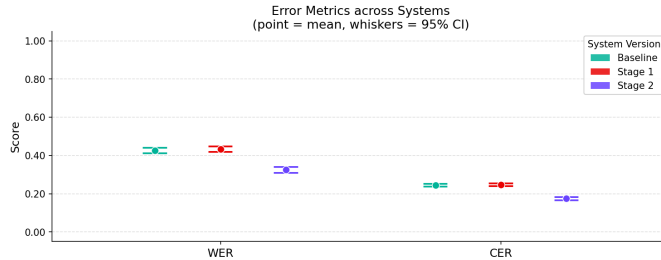


Fig. 2: Plots displaying word and character error rates across HaikuS2S stages. A lower score indicates greater correctness.

Stage	Emotional Similarity
Baseline	0.935 ± 0.0012
Stage 1	0.946 ± 0.0010
Stage 2	0.954 ± 0.0008

Table 1: Emotion similarity across HaikuS2S stages. A higher score indicates more similarity with the golden standard audio.

Naturalness and Speech Quality: All three systems (baseline, stage 1, and stage 2) had statistically significant differences in NISQA metrics. We see that the baseline and stage 1 consistently performed better than stage 2 (Fig. 4). We hypothesize that stage 2 performs poorly compared to the baseline and stage 1 because NISQA measures conversational naturalness, while stage 2 outputs more strictly haiku-structured audio. Since NISQA penalizes long pauses or unnatural conversational cadences and noise-level changes, which could include poetry line breaks, it is somewhat expected that stage 2 performs relatively poorly. **Chroma Alignment:** All three systems (baseline, stage 1, and stage 2) had statistically significant differences in chroma alignment. We see that stage 2 outperforms the baseline and stage 1 on two of the three metrics (Fig. 5). We also see that stage 1 performs equally or better compared to the baseline. We hypothesize that stage 1 is better than stage 2 on `chroma_stft` because this metric penalizes monotonicity, and we found that stage 2 outputted slightly more monotone audio outputs. We believe that the baseline consistently performs the worst because the baseline model is not meant to read poetry or prose. In contrast, stage 1 is trained on general poetry, and stage 2 is trained on poetry and also haiku with correct vocal tone, leading to higher similarity of output audios to gold standard audios.

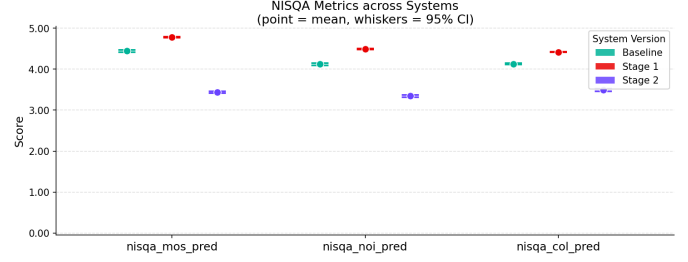


Fig. 3: Plots displaying NISQA metrics across HaikuS2S stages. A higher score indicates more natural-sounding speech.

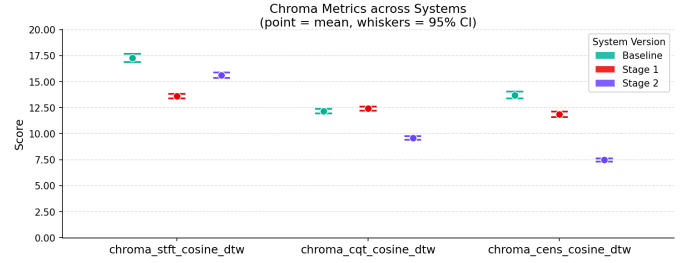


Fig. 4: Plots displaying chroma metrics across HaikuS2S stages. A lower score indicates more similarity with the golden audios.

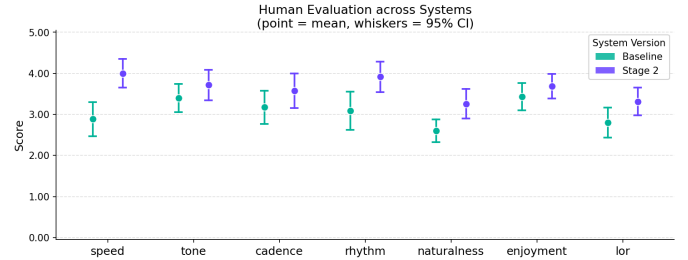


Fig. 5: Plots displaying human evaluation across HaikuS2S stages. A higher score indicates higher preference of system.

Human Evaluation: The baseline and stage 2 had higher scores on average across all evaluation metrics on a very small sample size of 35 (Fig. 6, 7). This is anticipated, as the baseline is not trained on poetry or haiku data, while stage 2 is trained on both. This leads to a higher human rating, as audio outputs are more similar to a recitation of a haiku by a human in stage 2 compared to the baseline. We argue that this result is important as this is the only metric we have that captures correct pronunciation and naturalness of poetry audios. **Overall Comparison:** We argue that stage 2 performs the best of the three systems. Combining intelligibility, tone, alignment, and human-preferred audios, we achieve our goal of creating a system that can correctly recite haiku. This performance increase is evident in both the metrics and the output audio for a human listener.

We also argue that stage 1 performs better than the baseline. Similarly, we see a notable improvement in the prosodic abilities of stage 2, which is clear to a human listener.

Future Work: In the future, it would be interesting to see if we can train the TTS to mimic the 5-7-5 constraints using G2P alignment decoding. We would also be interested in exploring the abilities of HaikuS2S on other kinds of poetry, such as sonnets and free-verse, prose, and general audiobooks. Further, we wish to develop a poetry-specific evaluation metric to augment or replace NISQA for more robust evaluation.

V. AI-GENERATED CONTENT DISCLOSURE

We used Gemini to generate our 1000 haiku (both training and testing data) used in the dataset 1000-haiku-audio. We also used Claude to aid in debugging fine-tuning scripts. No AI-generated content was used in the text, figures, or images included in this article.

VI. REFERENCES

- [1] RJ Skerry-Ryan, E Battenberg, Y Xiao, Y Wang, D Stanton, J Shor, RJ Weiss, R Clark, and RA Saurous, “Towards end-to-end prosody transfer for expressive speech synthesis with tacotron,” in *Proceedings of ICML*, 2018, arXiv:1803.09047.
- [2] Julia Koch, Florian Lux, Nadja Schauffler, Toni Bernhart, Felix Dieterle, Jonas Kuhn, Sandra Richter, Gabriel Viehhauser, and Ngoc Thang Vu, “PoeticTTS – controllable poetry reading for literary studies,” in *arXiv preprint arXiv:2207.05549*, 2022.
- [3] Ziyang Ma, Zhisheng Zheng, Jiaxin Ye, Jinchao Li, Zhifu Gao, Shiliang Zhang, and Xie Chen, “emotion2vec: Self-supervised pre-training for speech emotion representation,” *arXiv preprint arXiv:2312.15185*, 2023.
- [4] Giridhar Pamisetty and K Sri Rama Murty, “Prosody-tts: An end-to-end speech synthesis system with prosody control,” *Circuits, Systems, and Signal Processing*, vol. 42, no. 1, pp. 361–384, 2023.
- [5] Yi Lei, Shan Yang, Xinsheng Wang, and Lei Xie, “Msemotts: Multi-scale emotion transfer, prediction, and control for emotional speech synthesis,” *arXiv preprint arXiv:2201.06460*, 2022.
- [6] Guangyan Zhang, Ying Qin, Wenjie Zhang, Jialun Wu, Mei Li, Yutao Gai, Feijun Jiang, and Tan Lee, “iemotts: Toward robust cross-speaker emotion transfer and control for speech synthesis,” *arXiv preprint arXiv:2206.14866*, 2022.
- [7] Rodolfo Delmonte and Anton Maria Prati, “Sparsar: An expressive poetry reader,” in *Proceedings of the 14th EACL Demonstrations*, 2014, pp. 73–76.
- [8] X Wang et al., “Semantics for expressive tts,” in *ACL Workshop on Semantics for Expressive TTS*, 2015.
- [9] Anonymous, “Low resource emotional tts,” in *Interspeech 2020 Workshop*, 2020.
- [10] Peter Makarov, Ammar Abbas, et al., “Simple and effective multi-sentence tts with expressive and coherent prosody,” *arXiv preprint arXiv:2206.14643*, 2022.
- [11] S. Watanabe, H. Hori, S. Karita, et al., “ESPnet: End-to-end speech processing toolkit,” in *Interspeech*, 2018.
- [12] Siddhant Arora, Yifan Peng, Jiatong Shi, Jinchuan Tian, William Chen, Shikhar Bharadwaj, Hayato Futami, Yosuke Kashiwagi, Emiru Tsunoo, Shuichiro Shimizu, Vaibhav Srivastav, and Shinji Watanabe, “Espnet-sds: Unified toolkit and demo for spoken dialogue systems,” in *NAACL Demo*, 2025.
- [13] Yifan Peng, Shakeel Muhammad, Yui Sudo, William Chen, Jinchuan Tian, Chyi-Jiunn Lin, and Shinji Watanabe, “OWSM v4: Improving open whisper-style speech models via data scaling and cleaning,” in *Proceedings of the Annual Conference of the International Speech Communication Association (INTER-SPEECH)*, 2025.
- [14] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al., “Phi-3 technical report: A highly capable language model locally on your phone,” *Tech. Rep.*, Microsoft Research, 2024.
- [15] Tomoki Hayashi, Ryuichi Yamamoto, Katsuki Inoue, Takenori Yoshimura, Shinji Watanabe, Tomoki Toda, Kazuya Takeda, Yu Zhang, and Xu Tan, “Espnet-TTS: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit,” in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7654–7658.
- [16] Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchiani, Yu Zhang, Wei Han, and Ankur Bapna, “LibriTTS-R: A restored multi-speaker text-to-speech corpus,” May 2023.
- [17] Abubakar Abid, Ali Abdalla, Ali Abid, Dawood Khan, Abdulrahman Alfozan, and James Y. Zou, “Gradio: Hassle-free sharing and testing of ml models in the wild,” *ArXiv*, vol. abs/1906.02569, 2019.
- [18] Jiatong Shi, Jinchuan Tian, Yihan Wu, Jee-Weon Jung, Jia Qi Yip, Yoshiki Masuyama, William Chen, Yuning Wu, Yuxun Tang, Massa Baali, Dareen Alharthi, Dong Zhang, Ruifan Deng, Tejes Srivastava, Haibin Wu, Alexander Liu, Bhiksha Raj, Qin Jin, Ruihua Song, and Shinji Watanabe, “Espnet-codec: Comprehensive training and evaluation of neural codecs for audio, music, and speech,” in *2024 IEEE Spoken Language Technology Workshop (SLT)*, 2024, pp. 562–569.
- [19] Jiatong Shi, Hye jin Shim, Jinchuan Tian, Siddhant Arora, Haibin Wu, Darius Petermann, Jia Qi Yip, You Zhang, Yuxun Tang, Wangyou Zhang, Dareen Safar Alharthi, Yichen Huang, Koichi Saito, Jionghao Han, Yiwen Zhao, Chris Donahue, and Shinji Watanabe, “VERSA: A versatile evaluation toolkit for speech, audio, and music,” in *2025 Annual Conference of the North American Chapter of the Association for Computational Linguistics – System Demonstration Track*, 2025.

VII. APPENDIX: SUPPLEMENTARY TABLES

Stage	WER	CER
Baseline	0.425 ± 0.013	0.244 ± 0.007
Stage 1	0.433 ± 0.014	0.081 ± 0.007
Stage 2	0.324 ± 0.014	0.099 ± 0.008

Table 2: WER and CER across HaikuS2S stages.

Stage	MOS	NOI	COL
Baseline	4.442 ± 0.024	4.120 ± 0.028	4.126 ± 0.015
Stage 1	4.78 ± 0.011	4.490 ± 0.013	4.416 ± 0.005
Stage 2	3.439 ± 0.020	3.343 ± 0.023	3.488 ± 0.032

Table 3: NISQA metrics across HaikuS2S stages.

Stage	STFT	CQT	CENS
Baseline	17.276 ± 0.410	12.169 ± 0.219	13.710 ± 0.326
Stage 1	12.601 ± 0.232	12.403 ± 0.175	11.863 ± 0.266
Stage 2	15.623 ± 0.256	9.599 ± 0.176	7.466 ± 0.135

Table 4: Chroma metrics across HaikuS2S stages.

Stage	Speed	Tone	Cadence
Baseline	2.886 ± 0.415	3.400 ± 0.342	3.171 ± 0.405
Stage 2	4.000 ± 0.350	3.714 ± 0.373	3.571 ± 0.419
Rhythm	Naturalness	Enjoyment	LOR
3.086 ± 0.464	2.600 ± 0.280	3.429 ± 0.334	2.800 ± 0.366
3.914 ± 0.371	3.257 ± 0.362	3.686 ± 0.298	3.314 ± 0.338

Table 5: Human evaluation across HaikuS2S stages.