Controlling Difficulty of Generated Text for AI-Assisted Language Learning

Meiqing Jin*, **Liam Dugan***, **Chris Callison-Burch**
University of Pennsylvania
{mqjin, ldugan, ccb}@seas.upenn.edu

Abstract

Practicing conversations with large language models (LLMs) presents a promising alternative to traditional in-person language learning. However, most LLMs generate text at a near-native level of complexity, making them ill-suited for beginner learners (CEFR: A1–A2). In this paper, we investigate whether controllable generation techniques—specifically, modular methods that do not require model fine-tuning—can adapt LLM outputs to better support absolute beginners. We evaluate these methods through both automatic metrics and a user study with university-level learners of Japanese. Our findings show that while prompting alone fails to control output difficulty, the use of future discriminators (Yang and Klein, 2021) significantly improves output comprehensibility (from 40.4% to 84.3%). We further introduce a novel token-level evaluation metric, Token Miss Rate (TMR), that quantifies the proportion of incomprehensible tokens per utterance and correlates strongly with human judgments. To support future research in AI-assisted language learning, we release our code, models, annotation tools, and dataset.[1]

1 Introduction

Many learners struggle to acquire a second language (L2), often citing low motivation, anxiety, and a lack of conversational practice opportunities (Alzaanin, 2023; Papi and Khajavy, 2023). In parallel, large language models (LLMs) have emerged as fluent and engaging conversational agents (Tseng et al., 2024), prompting interest in their use for language learning. However, these models typically produce output at a near-native level, which can overwhelm beginners and impede learning.

Thus, a critical component of a second language conversation partner is **difficulty control**. The difficulty of utterances should fall within the *zone of proximal development*—not too easy but not too hard (Krashen, 1985). However, studies have shown that LLMs are unable to tailor their outputs to a specified difficulty level using simple prompting or in-context learning approaches (Imperial et al., 2023; Almasi et al., 2025; Ramadhani et al., 2023; Uchida, 2025; Benedetto et al., 2025; Zhang and Huang, 2024). Fine-tuning approaches offer more control but are often impractical due to cost, accessibility, or compatibility with closed-source models (Malik et al., 2024; Stowe et al., 2022).

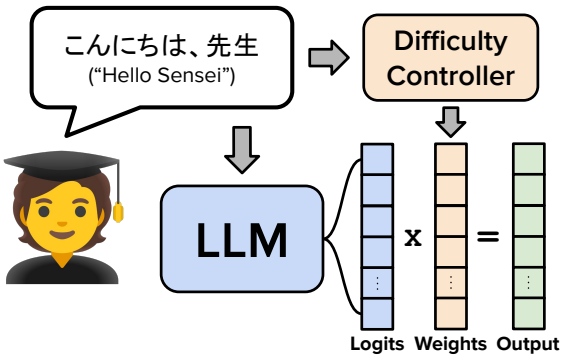


Figure 1: We control the difficulty of language model outputs to assist in language learning using future discriminators Yang and Klein (2021). We show that this works better than prompting in both automatic and human evaluations

In this work, we conduct the first study to evaluate LLM-based conversation partners with absolute beginner learners (CEFR: A1–A2). We focus on modular controllable generation techniques—that is, methods that operate externally to the model and do not require access to weights or training data. Specifically, we examine future discriminators (Yang and Klein, 2021), which bias generation toward desired characteristics during decoding.

Through automatic simulations and a user study with beginner-level Japanese learners, we find that future discriminators significantly enhance output comprehensibility while preserving fluency and

*Equal contribution
[1] https://github.com/EmmaJin0210/ChatLingual

naturalness. Compared to prompting alone, this approach more than doubles the rate of comprehensible utterances (from 40.4% to 84.3%).

In addition to these findings, we introduce the Token Miss Rate (TMR), a new metric that measures the percentage of tokens per utterance that are likely to be incomprehensible to the learner. TMR provides a finer-grained lens on model output and correlates strongly with human comprehensibility ratings, enabling scalable evaluation without relying solely on user studies.

Together, our results suggest that modular control techniques offer a practical path toward adapting LLMs for beginner language learners. By addressing difficulty control without retraining, and by rigorously evaluating output both holistically and at the token level, our work lays the foundation for more accessible, personalized AI-driven language tutoring systems.

2 Related Work

Language Learning with AI Chatbots Previous work on adapting AI for language learning have primarily focused on non-conversational settings. There have been studies using AI to generate example short stories (Malik et al., 2024), example sentences (Stowe et al., 2022; Glandorf and Meurers, 2024), recommendations for material at a learner's level (Jamet et al., 2024), and automated assessments (Caines et al., 2023). However, the potential for AI-based L2 conversation partners has been relatively under-explored.

Of the few studies that do evaluate AI for conversation-based language learning, most simply evaluate the vanilla model's ability with no explicit difficulty control applied (Lee et al., 2023; Wang, 2025; Hayashi and Sato, 2024; Almasi et al., 2025). The only previous work that evaluates the effect of difficulty control for AI conversation partners is Tyen et al. (2024) who evaluate a BlenderBot-2.7B model with and without a reranking-based difficulty control on a population of 160 L2 English speakers recruited through Prolific. While they found difficulty control largely ineffective, absolute beginners are very under-represented in their work with 0% of participants self-reporting as A1 and less than 12% reporting as A2. This is likely due to the recruitment material being in the target language. Our study is the first to specifically target these beginner speakers.

Controlling Difficulty of LLMs Previous work on using controllable generation techniques in L2 learning has primarily focused on re-ranking candidate generations at inference time. This involves training a classifier for the target difficulty and using it to rank a set of k candidate responses—outputting the one that maximizes the desired metric (Tyen et al., 2022; Jamet et al., 2024; Glandorf and Meurers, 2024; Malik et al., 2024). While this does improve LLMs' ability to adhere to a particular difficulty level, it is expensive and prone to failure in cases where the LLM produces no good candidates.

Stowe et al. (2022) attempt to fix this problem by fine-tuning on example utterances labeled with control tokens and Malik et al. (2024) additionally use reinforcement learning to encourage models to generate outputs at the correct difficulty level. While these techniques are effective at controlling difficulty, they are expensive and require all difficulty levels to be specified at training time. We are the first study to investigate the feasibility of using more modular controllable generation approaches on this task that do not require fine-tuning the full model.

3 Methods

In this section we give a brief introduction to controllable generation and explain the future discriminators method introduced by Yang and Klein (2021) as it is relevant for our comparison.

3.1 Controllable Text Generation

Controllable text generation is the task of generating text $X = x_1 \dots x_n$ conditioned on a specific desired attribute a (e.g. the desired difficulty of an utterance). Let $\mathcal{G}$ be an LLM which models $P(x_i|x_{1:i-1})$ for tokens $x_1 \dots x_n$. The likelihood of the full sequence $P(X)$ can be factorized as

$$P(X) = \prod_{i=1}^{n} P(x_i|x_{1:i-1})$$

To condition on a particular attribute, we must instead model $P(X|a)$, which modifies the previous factorization:

$$P(X|a) = \prod_{i=1}^{n} P(x_i|x_{1:i-1}, a)$$

Thus the controlled generation task can be simplified to finding a method for modeling the attribute-conditional distribution $P(x_i|x_{1:i-1}, a)$.

3.2 Future Discriminators (FUDGE)

In this work we employ Future Discriminators for Generation (FUDGE) (Yang and Klein, 2021) to accomplish the controllable generation task. FUDGE operates by approximating $P(x_i|x_{1:i-1}, a)$ based on a Bayesian factorization:

$$P(x_i|x_{1:i-1}, a) \propto P(a|x_{1:i}) \cdot P(x_i|x_{1:i-1})$$

Where $P(a|x_{1:i})$ denotes the likelihood of the attribute a given the prefix $x_{1:i}$. Intuitively this can be seen as a re-ranking of the candidate tokens from $\mathcal{G}$ by some attribute predictor model $\mathcal{M}$.

Importantly, this technique requires the predictor model $\mathcal{M}$ to model how likely it is that a particular sequence will have the attribute *in the future*. This is a very different task from the standard sequence classification task, thus preventing us from using off-the-shelf classifiers as predictors.

3.3 Training Predictors for Difficulty

Given a dataset D of text-attribute tuples $(x_{1:n}, a')$ the predictor model $\mathcal{M}$ is fine-tuned using all prefix-attribute pairs $(x_{1:i}, a')$ to learn the *future* likelihood of the attribute given the prefix. For our use-case we modify the predictor to minimize a multi-class cross entropy loss instead of a binary loss.

$$\mathcal{L}_{\mathcal{M}} = -\sum_{x} \log P(a'|x)$$

This encourages the model to output a distribution of likelihoods over n difficulty levels.

3.4 Inference

To re-rank candidate tokens for a particular prefix $x_{1:i-1}$ the predictor model is run on each potential next token sequence $x_{1:i}$. Doing this for the full distribution would be prohibitively expensive, we follow Yang and Klein (2021) and truncate the next token distribution to top-k=50 sampling.

We also employ a control parameter λ which allows us to modify the magnitude of the difficulty predictor. Thus the logit vector $\hat{y}$ for the full system is calculated by computing:

$$\hat{y} = \lambda a + (1 - \lambda)x$$

where x and a are the truncated logit vectors from the LLM and the predictor respectively. After this we normalize the vector $\hat{y}$ and sample the next token normally.

JLPT	CEFR	Example Conversation Topic
N5	**A1**	introduce yourself (such as your name, job/school, where you're from, etc.)
N4	**A2**	describe your favorite hobby and how often you do it
N3	**A2-B1**	talk about planning a birthday party: location, food, and guests
N2	**B1-B2**	describe a news story you found interesting, and why it caught your attention
N1	**B2-C1**	discuss recent advancements in regenerative medicine and their ethical implications in Japan

Table 1: JLPT levels and their corresponding CEFR levels (Japan Foundation, 2025) along with example conversation topics from the self-chat pipeline (§5.1).

4 Experimental Setup

In this section we discuss our evaluation metrics (§4.1), the data we use for training and evaluation (§4.2), and the models we use for comparisons (§4.3). In Section 5 we report the results for the automatic evaluation and in Section 6 we report the results of the human study. Additional detail on all aspects of our setup can be found in Appendix B.

4.1 Evaluation Metrics

Token Miss Rate (TMR) To evaluate the difficulty of the tutor's responses, we introduce a new metric we call **Token Miss Rate (TMR)**. This measures the percentage of tokens in a particular utterance that are above a user's specified level. We calculate this metric as follows:

$$TMR = \frac{\texttt{cnt\_above}}{\texttt{total\_tokens}}$$

Where `cnt_above` represents the number of tokens above a user's level and `total_tokens` represents the total number of tokens in an utterance. This intuitively measures the percentage of the output that is comprehensible and can serve as an automatic proxy for human judgements of understandability.

In order to map each token to its appropriate difficulty level we need to use whole word tokenization—not sub-word tokenization. To do this we use the Sudachi tokenizer (Takaoka et al., 2018) on segmentation mode C (coarse-grained) and set it to produce the base form (i.e. dictionary form) of each token (Table 2). This allows us to later classify each token to a JLPT level by exact match string searching over a list of vocabulary (see Appendix D).

ControlError In addition to our TMR metric we also calculate ControlError (Malik et al., 2024).

天気	が	いい	から	散歩	し	ましょう
天気	, が	, いい	, から	, 散歩	, する	, ます

Table 2: Example tokenization using Sudachi (Takaoka et al., 2018) on segmentation level C (coarse-grained). Verbs and grammatical structures are lemmatized for token matching and punctuation is removed.

This metric measures the squared distance between the predicted difficulty level of some input text x according to some classifier s and the target difficulty level t.

$$ControlError(x, t) = (s_{jlpt}(x) - t)^2$$

For our classifier model we use the fine-tuned Tokohu-BERT model[2] from Benedetti et al. (2024).

JReadability As a final metric for difficulty we calculate the JReadability score (Fujitani et al., 2012)—a Japanese-language analogue to the popular Flesch-Kincaid Grade Level metric (Kincaid et al., 1975). JReadability includes features such as sentence length, word difficulty, and syntactic complexity and was derived from statistical models trained on grade-level Japanese corpora.

Fluency Metrics We also report fluency metrics such as perplexity (PPL), average utterance length in tokens (Length), and average trigram diversity scores (div@3) for each output model. We calculate perplexity using the `Aya-Expanse 8B` model[3] from Dang et al. (2024) as it scored well on the Swallow open LLM leaderboard for Japanese (Okazaki et al., 2024).

4.2 Data

For our project we used two datasets. One for training the predictor model (§3.3) and one to provide JLPT vocabulary lists for evaluating Token Miss Rate (TMR) (§4.1).

Training Data For training the predictor model we use the **jpWaC-L** 1.0 corpus (Erjavec et al., 2008), a 300-million-token web corpus of sentences annotated with corresponding JLPT levels. The labels were derived using heuristics based on JLPT content specs (Japan Foundation, 2024).

[2] https://huggingface.co/bennexx/cl-tohoku-bert-base-japanese-v3-jlpt-classifier

[3] https://huggingface.co/CohereLabs/aya-expanse-8b

Evaluation Data For evaluation we use the `jlpt-anki-decks` dataset developed by chyyran, an open-source collection of Japanese vocabulary decks for the Anki spaced repetition platform (Chyyran, 2015). The vocabulary lists are based on the widely recognized compilations from `tanos.co.uk`, a foundational resource in JLPT preparation that provides curated vocabulary lists, grammar guides, and example sentences.

4.3 Comparisons

We implement the following four controllable generation methods as comparisons:

1. **Baseline Prompt**: A baseline prompt that instructs the tutor to only output at the target difficulty level (Almasi et al., 2025, *inter alia*).

2. **Detailed Prompt**: A more involved prompt that instructs the model using few-shot example conversations and vocab lists as well as more detailed descriptions of the target level.

3. **Overgenerate**: A re-ranking model following previous work (Tyen et al., 2022, *inter alia*) that re-ranks candidates based on estimated Token Miss Rate (TMR) calculated using vocabulary bins heuristically derived from the training data, as detailed in Section B.1.

4. **FUDGE**: A future discriminators-based method using the technique detailed in Section 3 from Yang and Klein (2021).

The prompts for the **Overgenerate** and **FUDGE** methods are identical to the baseline prompting method. Prompt templates for **Baseline** and **Detailed** can be found in Appendix B along with level and topic descriptions.

For all methods except FUDGE we evaluate with both `Qwen2.5-72B-Instruct` (Qwen et al., 2025) and `gpt-4-turbo` (OpenAI et al., 2024). Qwen2.5 was picked as the open-source model as it performed the best on the Swallow open LLM leaderboard for Japanese (Okazaki et al., 2024).

For the predictor model in FUDGE we used `ModernBERT` (Warner et al., 2024). We fine-tuned the model using a multi-class classification head through the Huggingface Trainer API for one epoch with a learning rate of $5e^{-5}$. More details on the training process can be found in Appendix A.

Figure 2: In the "self-chat" evaluation pipeline (§5.1) we evaluate our controlled generation methods by simulating conversations between a *student* LLM and difficulty-controlled *tutor* LLM. Tutor outputs are evaluated using **Token Miss Rate (TMR)** (§4.1) which quantifies the percentage of tokens in an utterance *above* the target level.

Automatic Evaluation Results						
Model	**Length**	**Avg. PPL ↓**	**div@3 ↑**	**JReadability ↑**	**TMR ↓**	**ControlError ↓**
Baseline (Qwen)	104.33	4.62	0.501	3.68	15.7	2.19
Baseline (GPT-4)	74.90	5.73	0.613	3.33	15.0	2.22
Detailed (Qwen)	127.90	**4.50**	0.555	3.59	14.4	1.89
Detailed (GPT-4)	73.45	5.67	**0.620**	3.39	13.7	1.92
Overgenerate (Qwen)	107.48	4.62	0.475	3.67	14.7	2.19
Overgenerate (GPT-4)	70.19	5.65	0.611	3.48	13.1	2.30
FUDGE ($\lambda = 0.25$)	77.81	4.87	0.580	3.74	14.2	2.15
FUDGE ($\lambda = 0.5$)	74.37	5.16	0.583	3.73	14.3	2.09
FUDGE ($\lambda = 0.8$)	75.80	5.13	0.599	3.78	13.3	1.89
FUDGE ($\lambda = 0.9$)	74.27	5.23	0.599	**3.82**	**11.9**	**1.78**

Table 3: Automatic evaluation of controlled generation approaches using the "self-chat" pipeline (§5.1). We see that FUDGE performs the best on difficulty control metrics such as Token Miss Rate (TMR) and ControlError while performing comparatively well on fluency metrics such as perplexity and diversity.

5 Automatic Evaluation

5.1 "Self-Chat" Pipeline

Following Almasi et al. (2025) we employ a self-chat approach to automatically evaluate our models. We simulate dialogues between a tutor agent and a student agent. The tutor is the subject of the evaluation, and uses controlled generation techniques (§4.3), while the student acts as the conversation partner. The Student's role is to simulate a language learner at a specific proficiency level and is intentionally set up without using any explicit difficulty control mechanisms. Both Student and Tutor have their own prompts that explain the scenario and instruct them to stay on topic.

To evaluate a given Tutor model's ability we simulate 15 dialogues per level—3 for each of the 5 levels of the Japanese Language Proficiency Test (JLPT) N1 through N5 (see Table 1). For each level, a set of three conversation topics are defined based on the topics of official JLPT sample exams. We do this to test the tutor's ability to control difficulty regardless of the student level or complexity of conversation topic.

5.2 Results

Difficulty Control In Table 3 we report the results of our automatic evaluation. Echoing previous work (Imperial et al., 2023), we see that prompting results in only modest improvements in Token Miss Rate (roughly -1.5%). We also see inconsistent gains from the Overgenerate approach. In both GPT-4 and Qwen models, overgenerate improves TMR over the baseline prompt but does not improve ControlError. This underscores the importance of ensuring high quality candidates for re-ranking. Finally, we see that FUDGE consistently outperforms all other methods achieving an 11.9% TMR with $\lambda = 0.9$ which is equivalent to a roughly 88% token comprehension rate.

Fluency Across all metrics we see that increasing the FUDGE control parameter λ increases the

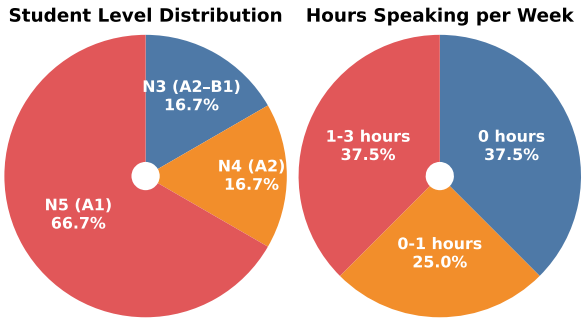


Figure 3: Distribution of JLPT / CEFR level of our study participants along with their self-reported average number of hours spoken per week.

difficulty control while keeping fluency metrics relatively stable. We see that on both trigram diversity (div@3) as well as perplexity (Avg. PPL) FUDGE performs comparatively well to other methods at high λ values. In our own testing we found that above $\lambda = 0.9$ the fluency of FUDGE begins to degrade. For our human study we selected $\lambda = 0.8$ as our comparison due to its perceived tradeoff between fluency and controllability.

6 Human Evaluation

In this section we discuss the results of our human evaluation. We conducted an in-lab user study to collect human judgments of the controllable generation methods we discuss in our work.

6.1 Participants

We recruited 6 participants from Japanese language courses at an R1 university in the United States. Since previous studies have struggled to include students of CEFR level A2 or lower (Tyen et al., 2024), we focus on lower levels of Japanese proficiency. Instructors of beginner- and intermediate-level courses shared our study materials via course platforms (e.g., Canvas) and email. To validate participants' language proficiency, we asked them to report their current Japanese class. We mapped their course enrollment to a JLPT level based on the syllabus and primary text, resulting in most participant proficiency levels between JLPT N5 and N4 (CEFR A1-A2) (see Figure 3).

6.2 Task Design

Each session lasted about 45 minutes and took place in a conference room on campus. After reviewing the consent form, we informed the participant that they would be interacting with four language learning chatbots through a voice-based web

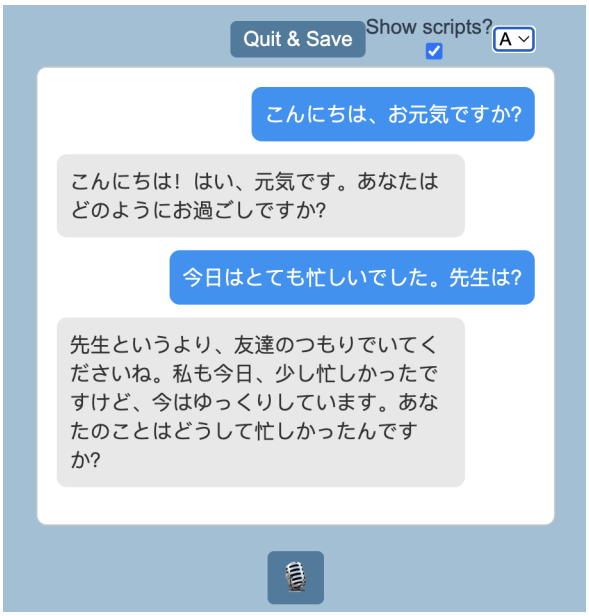


Figure 4: The voice-based interface used for the human evaluation. Users clicked the microphone icon when they wanted to speak and clicked again when finished.

interface. The participant then went through a three-turn tutorial conversation to become familiar with the interface. Their spoken input was transcribed with the API-based `whisper-1` model (Radford et al., 2022) and given as input to the backend model. The model's response was read aloud with audio generated by the OpenAI TTS-1 model[4]. For convenience, the text of the participant and model's responses were displayed on the screen.

Participants completed four 6-turn conversations[5], one for each of the controlled generation methods (`Baseline`, `Detailed`, `Overgenerate`, and `FUDGE`). For all four methods we used the `Qwen2.5-72B-Instruct` model as the chatbot. The order of the models was shuffled and hidden from participants, but the participants chose a new conversation topic each time from a list based on their JLPT level (see Appendix F.4).

6.3 Data Collection

Per-Turn Evaluation After each turn, participants were shown a transcript of the model's previous response and asked to review it word by word. They were instructed to highlight any spans of text that they did not understand, which we later used to compute TMR. Participants were also asked to

[4] http://platform.openai.com/docs/models/tts-1

[5] In the case of major transcription errors affecting fluency or comprehension, participants were allowed to continue for one additional turn.

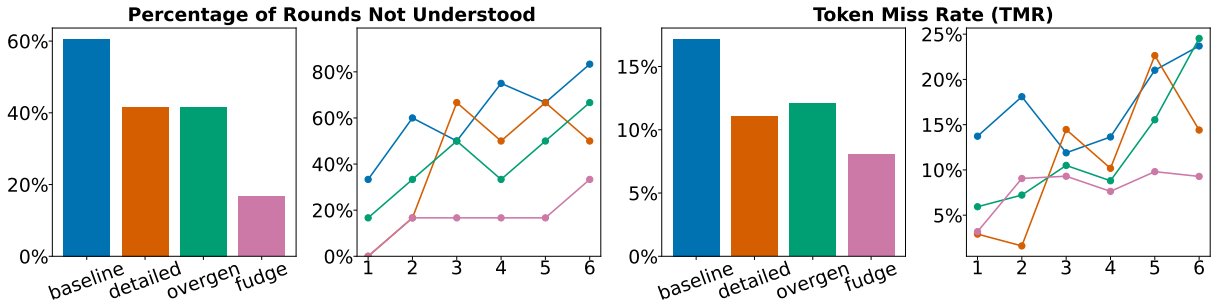


Figure 5: Results from the human evaluation for Baseline Prompt, Detailed Prompt, Overgenerate, and FUDGE. We see that FUDGE consistently has the lowest **TMR** (bar) and stays consistently low across multiple rounds (line).

Human Evaluation Results					
Model	**%Rounds Not Understood ↓**	**TMR ↓**	**Understandable? ↑**	**Effortful? ↓**	**Natural? ↑**
Baseline (Qwen)	60.6	17.2	5.17	5.17	5.67
Detailed (Qwen)	41.7	11.0	5.50	5.00	6.50
Overgenerate (Qwen)	41.7	12.1	6.33	5.83	**7.50**
FUDGE ($\lambda = 0.8$)	**16.7**	**8.0**	**7.67**	**4.67**	7.33

Table 4: Comparison of controlled generation approaches from the human evaluation using Qwen2.5-72B. We report Token Miss Rate (TMR) as well as average 1-10 Likert scores from post-session evaluation. We also report the percentage of rounds labelled as not understandable by the users. We see that high TMR is strongly correlated with a round being not comprehensible to human subjects ($\rho = 0.78$)

report whether they felt that they could understand the overall meaning of the response.

Per-Conversation Evaluation In addition to annotating each response, participants completed a brief questionnaire after each conversation, consisting of the following 10-point Likert-scale questions:

1. How much of the bot could you understand?
2. How effortful was it to talk to the bot?
3. How comfortable did you feel talking to the bot?
4. Did you feel like the bot's responses were natural given the context of the conversation?
5. Would you want to chat with this version of the bot again in the future?

The full text of the evaluation form can be found in Appendix F.

6.4 Results

In Table 4 we report the aggregated results of our human evaluation. Interestingly we see that the difference between FUDGE and other methods is much more pronounced in this setting. FUDGE achieves roughly half the TMR of the other methods and is the only method to come close to the oft-cited 95% comprehension threshold for full comprehension (Laufer, 1992).

We see this difference reflected most strongly in the Likert scale metrics (Figure 6). FUDGE was rated as the easiest to understand (7.67) and as taking the least amount of cognitive effort to speak with (4.67). We also see that FUDGE maintains a high degree of fluency, being rated the second most natural sounding (7.33) after the overgenerate method (7.50).

In addition to these aggregate results, in Figure 5 we show the plot of the TMR across subsequent round of conversation. Like Almasi et al. (2025) we also find that uncontrolled models suffer from **"alignment drift"**—gradually straying more and more from the target difficulty and getting higher TMR as a result. We find that FUDGE does a good job at maintaining a low TMR throughout multiple rounds of conversation and does not suffer from this alignment drift issue to the extent that other methods do. Future work should seek to further understand how these metrics change over time and how to ensure consistency in difficulty control over long conversations.

7 Conclusion

Learning a second language can open up significant opportunities. However, many beginner students find it difficult to practice conversational speech due to a lack of available conversation partners. In this paper we demonstrate the feasibility of using

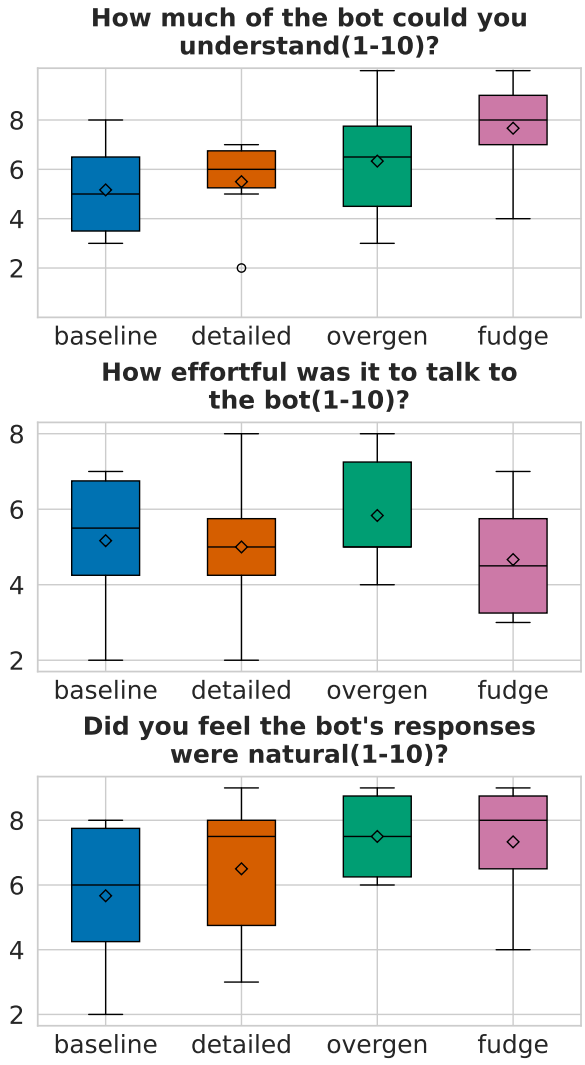


Figure 6: Our approach allows us to control the difficulty of the output of a language model to assist in language learning using future discriminators Yang and Klein (2021). We show that this works better than prompting in both automatic and human evaluations

controllable generation techniques along with state-of-the-art large language models as AI conversation partners for learning Japanese.

Through both human and automatic evaluations we show that using controllable generation techniques from Yang and Klein (2021) allows us to substantially constrain the difficulty of chatbot responses with over 83% of utterances rated as understandable by early and late beginner speakers (JLPT N4-N5; CEFR A1-A2). We also show that this result stays consistent across multiple rounds and that the outputs of the bot are natural.

Unlike prior work, our proposed technique does not require any fine-tuning of the base model—allowing for more personalized difficulty control. We imagine a scenario where students each have their own personal on-device predictor models trained to predict exactly their proficiency level. During conversation practice, these on-device predictors can combine with logits sent from an API-based chatbot for client-side difficulty control and personalization. We believe that our results constitute a meaningful step towards democratizing language learning and making conversation practice more accessible to speakers of all levels.

Limitations

We acknowledge that several sources of potential bias exist within this evaluation framework.

Definition of Difficulty It is hard to define an one-off approach to measure the level of difficulty of an utterance. Difficulty can be based on Vocabulary Level, Sentence Length, Grammatical Complexity, or Readability Scores, etc. Our criteria chosen or way of implementation may not perfectly align with a target learner's actual proficiency or capture the full nuance of language difficulty. However, despite this inherent subjectivity and potential misalignment, defining measurable difficulty criteria is a necessary first step to enable any form of automated difficulty control or evaluation, even if the criteria themselves are imperfect representations of human difficulty perception.

Undetected Tokens The Token Miss Rate is computed as the ratio of tokens detected in "above" levels to the total count of all tokens in the utterance. This calculation implicitly classifies tokens that were unable to be binned to a difficulty level as being understood by the learner. This can potentially skew the score for utterances with a high proportion of undetected tokens.

Acknowledgments

This research is supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the HIATUS Program contract #2022-22072200005. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Mina Almasi, Ross Deans Kristensen-McLachlan, and et al. 2025. Alignment drift in cefr-prompted llms for interactive spanish tutoring. *Preprint*, arXiv:2505.08351.

Eman Alzaanin. 2023. Efl learners' versus instructors' attributions of success and failure factors: A complexity theory perspective. *Language Teaching Research*, 0(0):13621688231189777.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Enrico Benedetti, Akiko Aizawa, and Florian Boudin. 2024. Automatically suggesting diverse example sentences for L2 Japanese learners using pre-trained language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 114–131, Bangkok, Thailand. Association for Computational Linguistics.

Luca Benedetto, Gabrielle Gaudeau, Andrew Caines, and Paula Buttery. 2025. Assessing how accurately large language models encode and apply the common european framework of reference for languages. *Computers and Education: Artificial Intelligence*, 8:100353.

Andrew Caines, Luca Benedetto, Shiva Taslimipoor, Christopher Davis, Yuan Gao, Øistein E. Andersen, Zheng Yuan, Mark Elliott, Russell Moore, Christopher Bryant, Marek Rei, Helen Yannakoudakis, Andrew Mullooly, Diane Nicholls, and Paula Buttery. 2023. On the application of large language models for language teaching and assessment technology. In *LLM@AIED*, volume 3487 of *CEUR Workshop Proceedings*, pages 173–197. CEUR-WS.org.

Chyyran. 2015. Jlpt anki decks. Accessed: 2025-05-17.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, Sandra Kublik, Meor Amer, Viraat Aryabumi, Jon Ander Campos, Yi-Chern Tan, Tom Kocmi, Florian Strub, Nathan Grinsztajn, Yannis Flet-Berliac, and 26 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *Preprint*, arXiv:2412.04261.

Tomaž Erjavec, Kristina Hmeljak Sangawa, and Yoshiko Kawamura. 2008. Japanese web corpus with difficulty levels jpWaC-l 1.0. Slovenian language resource repository CLARIN.SI.

Naoaki Fujitani, Jiyo Tsubaki, Kazuhide Yamamoto, and Manabu Okumura. 2012. Jreadability: A tool for assessing japanese text readability. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012): Demonstration Papers*, pages 103–110.

Dominik Glandorf and Detmar Meurers. 2024. Towards fine-grained pedagogical control over English grammar complexity in educational text generation. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 299–308, Mexico City, Mexico. Association for Computational Linguistics.

Kotaro Hayashi and Takeshi Sato. 2024. The effectiveness of chatgpt in enhancing english language proficiency and reducing second language anxiety (l2). In *WorldCALL Official Conference Proceedings*, pages 201–208.

Joseph Marvin Imperial, Harish Tayyar Madabushi, and et al. 2023. Flesch or fumble? evaluating readability standard alignment of instruction-tuned language models. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 205–223, Singapore. Association for Computational Linguistics.

Henri Jamet, Maxime Manderlier, Yash Raj Shrestha, and Michalis Vlachos. 2024. Evaluation and simplification of text difficulty using llms in the context of recommending texts in french to facilitate language learning. In *Proceedings of the 18th ACM Conference on Recommender Systems*, RecSys '24, page 987–992, New York, NY, USA. Association for Computing Machinery.

Japan Foundation. 2024. Japanese-Language Proficiency Test: Test Guide. Accessed: 2025-05-17.

Japan Foundation. 2025. Indication of the CEFR Level for Reference. Accessed: 2025-05-17.

J. P. Kincaid, R. P. Fishburne, R. L. Rogers, and B. S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical Report Research Branch Report 8-75, Naval Technical Training Command, Naval Air Station Memphis, Millington, TN.

Stephen D. Krashen. 1985. *The Input Hypothesis: Issues and Implications*. Longman, New York.

Batia Laufer. 1992. *How Much Lexis is Necessary for Reading Comprehension?*, pages 126–132. Palgrave Macmillan UK, London.

Seungjun Lee, Yoonna Jang, Chanjun Park, Jungseob Lee, Jaehyung Seo, Hyeonseok Moon, Sugyeong Eo, Seounghoon Lee, Bernardo Yahya, and Heuiseok Lim. 2023. PEEP-talk: A situational dialogue-based chatbot for English education. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 190–207, Toronto, Canada. Association for Computational Linguistics.

Ali Malik, Stephen Mayhew, Christopher Piech, and Klinton Bicknell. 2024. From tarzan to Tolkien: Controlling the language proficiency level of LLMs

for content generation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15670–15693, Bangkok, Thailand. Association for Computational Linguistics.

Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano, Hideki Matsuda, Kazutoshi Takaoka, and Masayuki Asahara. 2000. Japanese morphological analysis system chasen version 2.0 manual. In *Proceedings of the Workshop on Usage of Language Resources at the Third International Conference on Language Resources and Evaluation (LREC)*, Athens, Greece. European Language Resources Association (ELRA).

Naoaki Okazaki, Sakae Mizuki, Youmi Ma, Koki Maeda, Kakeru Hattori, Masanari Ohi, Hinari Shimada, Taihei Shiotani, Koshiro Saito, Rio Yokota, Kazuki Fujii, Taishi Nakamura, Takumi Okamoto, Ishida Shigeki, Yukito Tajima, Masaki Kawamura, and Hiroya Takamura. 2024. Swallow project leaderboard. https://swallow-llm.github.io/evaluation/index.en.html?index=%22__ALL__%22. GitHub repository available at https://github.com/swallow-llm/leaderboard.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Mostafa Papi and Hassan Khajavy. 2023. Second language anxiety: Construct, effects, and sources. *Annual Review of Applied Linguistics*, 43:127–139.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *arXiv preprint*.

Reski Ramadhani, Hilmi Aulawi, and Risma Liyana Ulfa. 2023. Readability of reading texts as authentic materials issued by chatgpt: A systemic functional perspective. *Indonesian Journal of English Language Teaching and Applied Linguistics*, 8(2):149–168.

Kevin Stowe, Debanjan Ghosh, and Mengxuan Zhao. 2022. Controlled language generation for language learning items. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 294–305, Abu Dhabi, UAE. Association for Computational Linguistics.

Kazuma Takaoka, Sorami Hisamoto, Noriko Kawahara, Miho Sakamoto, Yoshitaka Uchida, and Yuji Matsumoto. 2018. Sudachi: a japanese tokenizer for business. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Yu-Min Tseng, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, and Yun-Nung Chen. 2024. Two tales of persona in LLMs: A survey of role-playing and personalization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16612–16631, Miami, Florida, USA. Association for Computational Linguistics.

Gladys Tyen, Mark Brenchley, Andrew Caines, and Paula Buttery. 2022. Towards an open-domain chatbot for language practice. In *Proceedings of the 17th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2022)*, pages 234–249, Seattle, Washington. Association for Computational Linguistics.

Gladys Tyen, Andrew Caines, and Paula Buttery. 2024. LLM chatbots as a language practice tool: a user study. In *Proceedings of the 13th Workshop on Natural Language Processing for Computer Assisted Language Learning*, pages 235–247, Rennes, France. LiU Electronic Press.

Satoru Uchida. 2025. Generative ai and cefr levels: Evaluating the accuracy of text generation with chatgpt-4o through textual features. *Vocabulary Learning and Instruction*, 14(1):2078.

Ying Wang. 2025. A study on the efficacy of chatgpt-4 in enhancing students' english communication skills. *SAGE Open*, 15(1):21582440241310644.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint*, arXiv:2412.13663.

Kevin Yang and Dan Klein. 2021. FUDGE: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, Online. Association for Computational Linguistics.

Zhihui Zhang and Xiaomeng Huang. 2024. The impact of chatbots based on large language models on second language vocabulary acquisition. *Heliyon*, 10(3):e25370.

Andrew Zhu, Liam Dugan, Alyssa Hwang, and Chris Callison-Burch. 2023. Kani: A lightweight and highly hackable framework for building language

Hyper-parameter	Value
per_device_train_batch_size	16
per_device_eval_batch_size	16
num_train_epochs	3
learning_rate	5×10^{-5}
evaluation_strategy	epoch
logging_steps	10
dataloader_num_workers	4

Table 5: Hyper-parameters used to train the FUDGE predictor model.

model applications. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 65–77, Singapore. Association for Computational Linguistics.

A Hyperparameters & Configurations

This appendix outlines the training configuration used for our FUDGE predictor model. We finetune the `answerdotai/ModernBERT-base` transformer on sentence-level JLPT annotations, using a weighted sampling strategy to address class imbalance. All relevant hyperparameters and settings are summarized in Figure 5.

B Implementation Details

In this section we will go over more details about how our multi-modal AI tutor was constructed.

Kani To build our chatbot we use the Kani[6] framework, which is a flexible abstraction for building multi-turn conversational agents in Python (Zhu et al., 2023). It provides lightweight abstractions around LLM engines (`BaseEngine`), message formats (`ChatMessage`, `ChatRole`), and generation management (`Completion`). This allows us to easily swap between local and API based chat models without changing our interface code. In addition, we implement the controllable generation using the HuggingFace `LogitsProcessor` interface which allows us to use our predictor model with any valid Huggingface decoder-only language model.

Hardware For inference we used six NVIDIA RTX A6000 GPUs queried via local server through a web socket. We allocated one machine to the predictor model and the other five machines to host the 72 billion parameter Qwen model. For cases during "self-chat" where a student Qwen agent was chatting with a tutor Qwen agent, we loaded one instance of the model and swapped out the chat context as necessary.

[6] https://github.com/zhudotexe/kani

B.1 Heuristic Re–Ranking for the Overgenerate Method

The Overgenerate method follows a *generate–then–rank* paradigm. Given a user prompt, the underlying HF model (GPT-4 / Qwen) first produces $N = 5$ candidate continuations.

B.1.1 Vocabulary binning for Overgenerate.

Step 1: Tokenization. For each collection of sentences belonging to a certain JLPT level in the jpWaC dataset, we tokenize every line by using SudachiPy in coarse-grained mode C. Tokenization includes lemmatization and punctuation removal, and we additionally filter for strings consisting entirely of Japanese script (kanji, hiragana, katakana) using a Unicode whitelist.

Step 2: Frequency Aggregation. We keep track of two frequencies globally:

- how often each token appears in that level's subcorpus
- per-level token occurrences across all levels

Step 3: Rare Word Filtering. To eliminate spurious and rare tokens, we apply two threshold filters:

- **Global frequency filter:** words must appear more than 10^{-6} of the total tokens across all levels;
- **Level-specific filter:** words must appear more than 10^{-6} of the tokens within their level.

These thresholds eliminate noise and ensure that bin assignments are based on consistent patterns.

Step 4: Best-Level Assignment. We assign each token w to the easiest level in which it reaches a sufficient relative frequency. That is, for each token, we scan levels N5→N1 in order and assign it to the first level where its frequency is non-trivial relative to that level's total token count:

$$\text{score}_{w,\ell} = \frac{\text{count}_{w,\ell}}{\text{total tokens}_{\ell}}$$

The first level ℓ for which $\text{score}_{w,\ell}$ exceeds a small threshold is selected as the word's bin.

Step 5: Extracting Unique Bag-of-Words. To reduce overlap between levels, we construct disjoint vocabulary bins by progressively subtracting already-assigned tokens from lower levels. This results in a set of per-level vocabulary lists where each word appears only in one level.

JLPT N5	36563
JLPT N4	103298
JLPT N3	372421
JLPT N2	137312
JLPT N1	2627335
Total	3276929

Table 6: Total number of sentences per JLPT level in the jpWaC corpus.

Outputs. Each final vocabulary bin is written as a plain-text file named `{level}_bagofwords.txt`, which contains one token per line. These lists are used in the Overgenerate engine for detecting level-appropriate vs. above-level vocabulary in generated responses.

Re-ranking. We evaluate every candidate in parallel threads. Candidates are first sorted by their TMR (ascending) and then, to break ties, by total token count (shorter is preferred). The top-ranked continuation is returned to the user and logged together with its diagnostic statistics.

C Extra Dataset Details

In this section we provide additional details about the datasets used in the project for training and evaluating the system.

C.1 jpWaC

The corpus was collected using WaCkY tools (Baroni et al., 2009), lemmatized and POS-tagged via Chasen (Matsumoto et al., 2000), with tags mapped to both English and Japanese tagsets. Each sentence includes a `@level` attribute indicating overall difficulty, enabling fine-grained control over input complexity for language modeling tasks. This dataset is used as the sentence and vocabulary pool of the Japanese language for our system, used for training our FUDGE predictor, as well as for the prompting and overgeneration baseline methods mentioned in Section 4.3. The dataset's distribution of sentences across different JLPT levels is shown in Table 6.

D JLPT Vocabulary Bins

D.1 Anki Deck Parsing

To construct vocabulary bins grouped by JLPT level, we extract and normalize vocabulary entries from (Chyyran, 2015), originally provided in the form of '.apkg' files. These decks are organized by JLPT level (N5–N1) and contain Japanese expressions, readings, and glosses. Our parsing process extracts useful vocabulary while handling variations in formatting and ambiguity in expression/reading pairs.

Input Format. Each JLPT level deck is imported into Anki and stored as a local SQLite file under 'collection.anki2'. Each vocabulary entry is stored as a row in the 'notes' table, where the 'flds' field contains tab-separated content fields, including: - Expression (kanji, kana, or mixed), - Reading (optional, typically kana), - Gloss/Meaning (in English).

Parsing Overview. We use a Python script to:

1. Traverse the anki deck for each JLPT level,
2. Parse entries from the 'notes' table,
3. Clean and normalize parentheses and tilde variations in expressions/readings,
4. Expand expressions and readings that include alternations or parenthetical annotations,
5. Perform heuristic-based filtering to exclude duplicates or overly ambiguous entries,
6. Generate one flat JSON vocabulary file per JLPT level, saved to disk for later use.

Normalization Heuristics. We apply several normalization steps:

- Japanese parentheses are first converted to standard parentheses (and).
- Tilde-like characters (e.g., ～) are stripped entirely, as they denote alternation or ellipsis.
- Parenthetical expressions (e.g., （す る), （を）) are expanded using two strategies:
  - The outside form (e.g., 話す (こと) becomes 話す)
  - The full form with parenthetical inserted (e.g., 話すこと)
- Readings are similarly expanded, and matched against expressions using filters to avoid over-generation.

JLPT N5	introduce yourself (such as your name, job/school, where you're from, etc.) describe what you usually do in the morning and evening talk about your favorite food and where you usually eat it
JLPT N4	explain what you will do this weekend and with whom describe your favorite hobby and how often you do it talk about a typical day at school or work, including schedule and people you meet
JLPT N3	describe a travel experience: where you went, what you saw, and who you went with talk about planning a birthday party: location, food, and guests describe your favorite movie: the story, characters, and why you like it
JLPT N2	describe a recent news story you found interesting, and why it caught your attention explain one cultural difference between Japan and your country, and how it affects communication discuss a challenge people face when communicating in a Japanese workplace
JLPT N1	discuss recent advancements in regenerative medicine and their ethical implications in Japan explain the role of quantum computing in future communication technologies and how Japan is preparing for it analyze the impact of declining biodiversity on Japan's agricultural sustainability and food security

Table 7: The full list of conversation topics used in our self-chat evaluation pipeline (see Section 5.1)

```
You are roleplaying as a student learning {language} at the {level_word} level.
You are having a conversation with your language partner (i.e. the user) to practice
    {language}.
The topic of this conversation is: {topic}.
As a {level_word} student, you are: {desc}.
You must speak using only the vocabulary and grammar allowed at this level.
You are not in a formal class - this is casual language practice with someone your
    age.

You should ALWAYS follow the rules below:
1. You should stick to using only the vocabulary and grammar allowed at your level
    mentioned above.
2. Do not ask the user to teach you things. Just bring up the topic naturally and
    continue the conversation.
3. Your conversation should revolve around the topic of: {topic}. Respond one idea at
    a time.
4. You must keep the conversation going. Do not assume the conversation is over just
    because a few turns have passed.
5. DO NOT say anything like 'goodbye', 'see you next time', or anything else that
    signals the end of this conversation. You MUST keep the conversation going.
6. You should speak in {language} and {language} only.
```

Figure 7: System prompt defining the Student model.

Filtering Strategy. The script includes several regular expression-based heuristics to exclude non-informative entries:

- Skip readings that are only one hiragana character (e.g., の), or extremely short forms unlikely to be useful alone.
- Avoid adding a reading as a standalone entry if it overlaps directly with another alternative expression.
- Skip duplicates, malformed entries, or entries missing either expression or meaning.

Output Format. The result is a set of per-level JSON files, e.g., `n5.json`, each containing normalized entries in the form:

```
{
  "会う": {
    "meaning": "to meet, to see"
  },
  "あなた": {
    "meaning": "you"
  },
  ...
}
```

These files are used downstream for populating vocabulary bins, level-appropriate conversation prompts, and difficulty evaluation.

In Table 8, we show a representative sample of vocabulary items used at each JLPT level from N5 (easiest) to N1 (hardest), along with their English glosses.

```
You are a {language} language tutor.
Your goal is to help the user improve their {language} conversation skills through a
    natural, back-and-forth dialogue.
You are a native {language} speaker, around the same age as the user, and you're
    acting as their language partner.
The user you are speaking with is at the {level_word} level.
Please be aware of the user's level at all times and ensure that all of your
    responses stay within a level that is understandable to a user at this
    proficiency.
Stick to the topic the user brings up. Do not suggest topics or introduce new topics
     on your own.
Stay on the user's topic and follow their lead throughout the conversation.
Don't pick on small mistakes the user makes. If the user makes a really big grammar
    mistake, remind the user by saying the corrected version of the sentence. DO NOT
     try to explain their mistake.
You should keep the conversation going back and forth.
You must never say things like 'goodbye', 'see you tomorrow', or anything else that
    signals the end of the conversation unless the user initiates it.
You should speak in {language} and {language} only.
```

Figure 8: System prompt defining the Tutor model.

JLPT N5	明日 (tomorrow), あなた (you), 魚 (fish), いいえ (no, not at all), 少し (little, few), 有名 (famous) 先生 (teacher, professor; master; doctor), 有る (to be, to have), 会う (to meet, to see), 行く (to go)
JLPT N4	生きる (to live), 心配 (worry, concern), 始める (to start, to begin), 止める (to end, to stop) 中学校 (junior high school), 会話 (conversation), そろそろ (gradually, soon), 専門 (major; speciality)
JLPT N3	様々 (varied, various), 全て (all, the whole, entirely), 集まり (gathering, meeting, collection) 印象 (impression), 作品 (work, opus, production), わざと (on purpose), 変化 (change, variation, shift)
JLPT N2	重力 (gravity), 純粋 (pure, genuine, unmixed), 先祖 (ancestor), 課税 (taxation), 清い (clear, pure, noble) 強化 (strengthen, intensify, reinforce), 論ずる (to argue, to discuss), 必需品 (necessities, essential)
JLPT N1	賢明 (wisdom, intelligence, prudence), 倹約 (thrift, economy, frugality), 鉱業 (mining industry) 護衛 (guard, convoy, escort), 戸籍 (census, family register), 臆病 (cowardice, timidity), 放射能 (radioactivity)

Table 8: Example vocabulary at each JLPT level.

E Web Interface

Participants engaged with a custom-built web interface for the human evaluation. Figure 10 presents the stages of preparation for each round of user study, including navigating from the homepage (a), selecting the corresponding JLPT level of the user (b), navigating to the chat interface (c), and selecting which method to run on (d). We complete all setup before each round before we hand the laptop to the participant to begin their conversation with the bot.

F Human Study Details

F.1 Intake Form

After signing up and prior to their user study session, each participant completed an intake form. This form collected language background, class enrollment, and JLPT experience, along with confirming informed consent for participation and recording. The full content of the intake and consent form is shown in Figure 11.

F.2 Mapping of Textbook to JLPT level

To estimate a participant's JLPT level, we mapped their current university Japanese course (as self-reported) to an approximate JLPT level based on textbook progression and kanji coverage. Table 9 summarizes this mapping, which uses both Genki and Tobira textbooks as primary anchors for N5 through N3 levels, and more advanced materials such as news articles to the N2 level.

F.3 Annotation Interface

Within a user study session, the participant has a total of four full conversations of six rounds back-and-forth. After each conversation, participants were asked to review the system's output and annotate any words or phrases they could not understand. They were also asked to indicate whether they understood the overall meaning of the round. At each round, the system's generation is sent to an iPad

```
You are a {language} language tutor.
Your goal is to help the user improve their {language} conversation skills through a
    natural, back-and-forth dialogue.
You are a native {language} speaker, around the same age as the user, and you're
    acting as their language partner.
The user you are speaking with is at the {level_word} level.
This means that they: {level_description}.
An example of a short dialogue at the user's comprehension level is:
{level_conv_example}

Please be aware of the user's level at all times and ensure that all of your
    responses stay within a level that is understandable to a user at this
    proficiency.

You should ALWAYS follow the rules below:
1. {level_guidelines}
2. Remember, the user is a language learner, not a native speaker. You should make
    sure that you are speaking in a way that the user could understand with their
    current {language} level.
3. You should try to match the user's abilities of understanding and speaking: if
    the user only uses simple expressions, you should only use simple expressions as
     well.
4. During the conversation, don't pick on small mistakes the user makes. If the user
     makes a really big grammar mistake, remind the user by saying the corrected
    version of the sentence. DO NOT try to explain their mistake.
5. Stick to the topic the user brings up. Do not suggest topics or introduce new
    topics on your own. Stay on the user's topic and follow their lead throughout
    the conversation.
6. You should keep the conversation going back and forth.
7. You must never say things like 'goodbye', 'see you tomorrow', or anything else
    that signals the end of the conversation unless the user initiates it.
8. You should speak in {language} and {language} only.
9. Here are some expressions the user knows: {known_expressions}. Restrict your
    speaking to use these words and other words of similar or lower difficulty.
```

Figure 9: System prompt defining the Tutor model (detailed prompt).

Genki I, **Genki II** (Lesson 13-Lesson 14)	JLPT N5
Genki II, **Tobira: Gateway to Advanced Japanese** (Unit 1–Unit 2)	JLPT N4
Tobira: Gateway to Advanced Japanese (Unit 9-Unit 14)	JLPT N3
Advanced materials selected from the internet, newspapers, and books	JLPT N2

Table 9: Mapping from textbooks used in university Japanese courses to their corresponding JLPT levels.

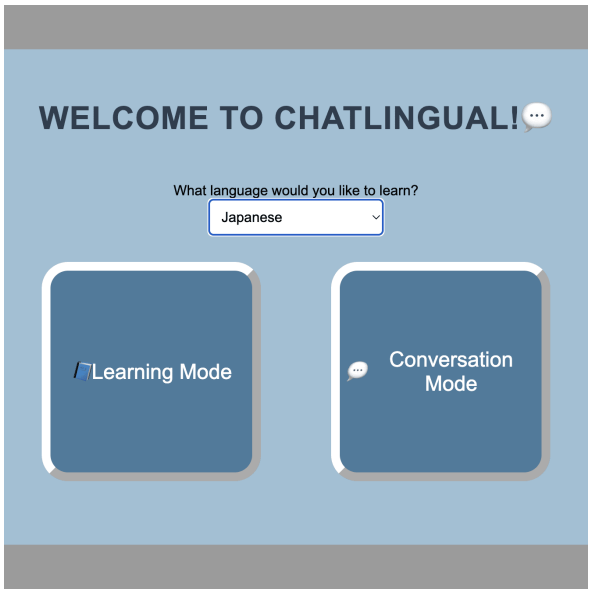
by email for prompt highlighting after each and every round. As shown in Figure 12, red highlights denote individual words and phrases marked as incomprehensible, while green or red dots indicate binary comprehension of the round as a whole.

F.4 Conversation Topics

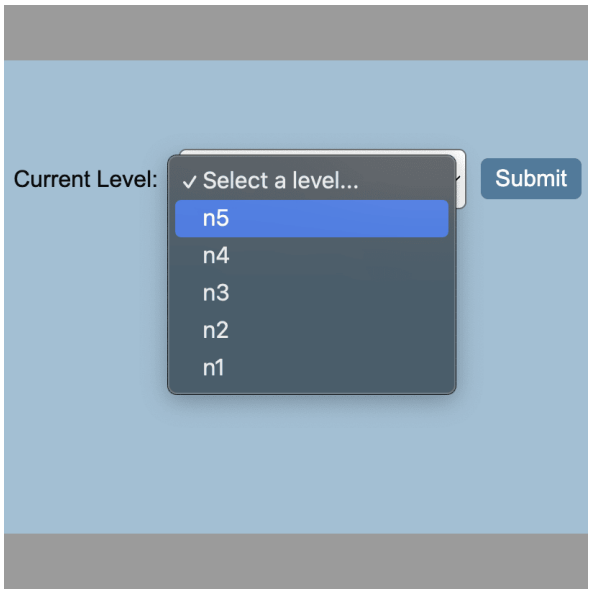
Based on each participant's most recent Japanese course and their estimated JLPT level, we compile a list of level-appropriate conversation topics utilizing the corresponding textbooks of each level (see Figure 13). Before each round, the system randomly selects three topics from the corresponding list and presents them for the participant to choose from.

G Use of AI Assistants

AI assistants were used in writing this paper to enhance clarity of wording, format tables and figures, and help fix compilation errors. The models used were ChatGPT and GPT-o3.

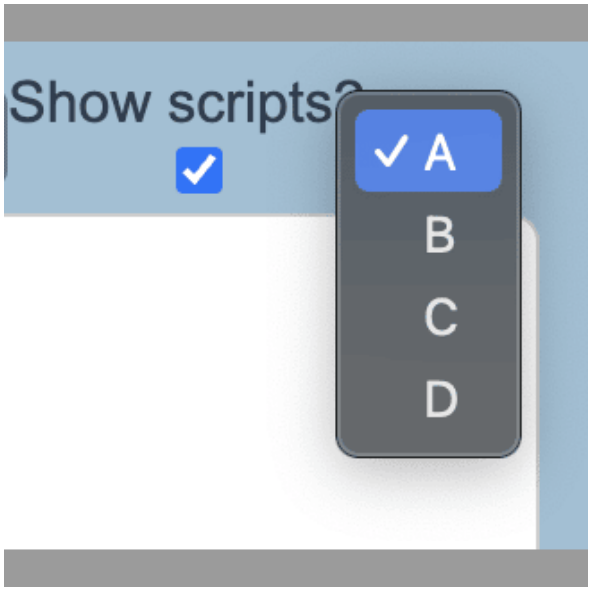


(a) The **homepage** of the interface used for the human evaluation. For the purpose of this study, we only use the Conversation Mode.



(b) We select the **user's corresponding level** through a dropdown.

(c) The user interacts with the **main chat interface**, which is initially blank with a recording button.



(d) We select the **method** through a dropdown before each round; we also set **show scripts** to true so that the user can read the text output of the bot.

Figure 10: Preparation on interface for each user study conversation round: Homepage (a), Level-Selection (b), Chat Interface (c), and Method-Selection (d).

Intro:
Thank you again for signing up to participate in our study!

This is a small-scale user study, so your participation truly means a lot to us.
The information you provide here will help us schedule your session, accommodate your preferences, and better understand your language background.

All responses will be kept confidential and used solely for research purposes.
We are collecting your email only to coordinate your session and follow up if needed. It will not be shared or used for any other purpose.

Informed Consent:

1. Informed Consent Statement:
This study involves interacting with an AI-powered chatbot in Japanese. While we've designed it to be level-appropriate,
the chatbot may occasionally provide inaccurate or confusing responses.
Your interactions will be recorded and anonymized for research purposes.
Participation is voluntary, and you may withdraw at any time without penalty.

All data collected will be kept confidential and used solely for research purposes.
No personally identifiable information will be shared outside the research team.
Audio from your session will be used to analyze the interaction and will be deleted after it is transcribed and anonymized.

Consent Confirmation:
By continuing, you confirm that:
- You are at least 18 years old
- You have read and understood the information above
- You voluntarily consent to participate in this study

[ ] I consent to participate in this study under the terms described above.

Participant Background:

1. Did you grow up speaking or hearing Japanese at home as a child?
[Yes / No]

2. Approximately how many hours per week do you spend speaking Japanese conversationally outside of class?
[Text response]

3. Approximately how many hours per week do you spend listening to conversational Japanese (podcasts, etc.) outside of class?
[Text response]

4. If you're currently taking a Japanese class at the university, which class are you enrolled in?
[Text response]

5. How many semesters of Japanese study have you completed?
[Text response]

6. Have you taken the Japanese Language Proficiency Test (JLPT)?
[Yes / No]

If yes, what level did you achieve?
[N5 / N4 / N3 / N2 / N1]

If not, what level do you estimate you're at (based on textbook, course, or self-assessment)?
[N5 / N4 / N3 / N2 / N1]

Figure 11: Full content of the form used to collect intake and consent data in the study.

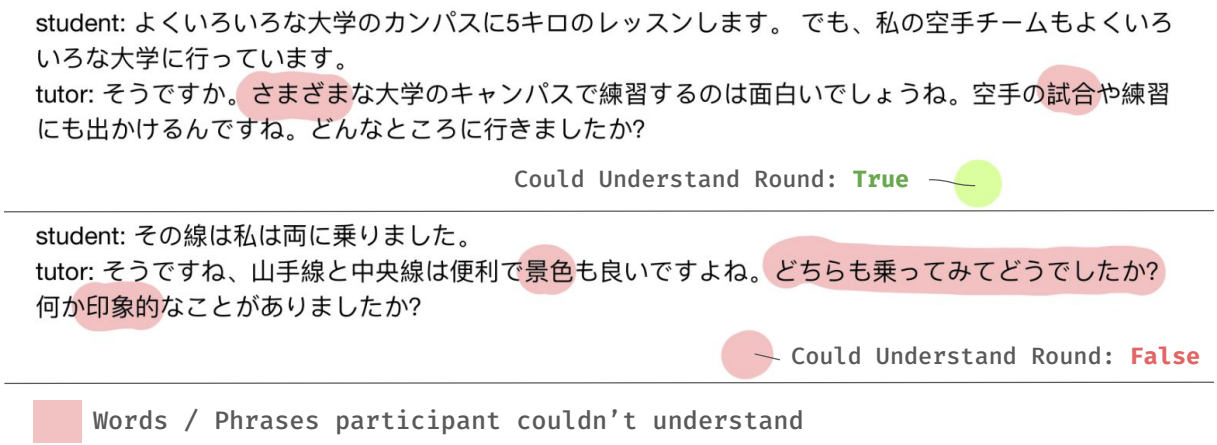


Figure 12: The annotation interface used during human evaluation. After each round, participants were asked to highlight specific words or phrases they did not understand (shown in red). They also indicated whether they understood the overall meaning of the round: a green dot denotes comprehension, and a red dot denotes lack of understanding.

JLPT N5 Topics List:
- Summer vacation plans
- Weekend hobbies or routines
- Favorite movie or TV show
- Favorite book, story, or folklore
- Favorite sport or physical activity
- A memorable trip or vacation
- A time you got sick
- A favorite holiday or festival

JLPT N4 / Early N3 Topics List:
- A time something was stolen
- A time you were hurt or injured
- Doing house chores
- A habit that annoys you
- A time you reported a crime or accident
- A favor you asked from someone
- A time you had to say goodbye
- A promise or decision you made
- A future goal that you have
- A region in Japan you want to visit
- A famous place you've been to
- A local food or specialty you like
- A festival you've attended or want to see
- Your hometown and what it's known for
- A memorable travel story
- A seasonal event you enjoy
- A travel recommendation for a friend

JLPT N2 Topics List:
- Describe a recent news story you found interesting, and why it caught your attention
- Explain one cultural difference between Japan and your country, and how it affects communication
- Discuss a challenge people face when communicating in a Japanese workplace
- Talk about a social issue you care about and why it's important to you
- Describe a time you had to be polite in a difficult situation
- Compare education systems in Japan and your home country
- Share your opinion on using AI or technology in daily life
- Describe a tradition or custom from your country and how it's changing
- Talk about how your communication style changes depending on the situation
- Discuss the pros and cons of working remotely or studying online
- Talk about a piece of Japanese literature you like
- Discuss how Japanese society is addressing the social issue of aging population

Figure 13: Conversation topics participants chose from, grouped by estimated corresponding JLPT level.