

Data-efficient Targeted Token-level Preference Optimization for LLM-based Text-to-Speech

Rikuto Kotoge^{1,2}, Yuichi Sasaki^{1,3}

¹SpiralAI Inc., ²The University of Osaka, ³Shizuoka University

Correspondence: r_kotoge@go-spiral.ai

Abstract

Aligning text-to-speech (TTS) system outputs with human feedback through preference optimization has been shown to effectively improve the robustness and naturalness of LLM-based TTS models. Current approaches primarily require paired desirable and undesirable samples at the utterance level. However, such pairs are often limited in TTS output data, and utterance-level formulation prevents fine-grained token-level optimization needed for accurate pronunciation alignment. In this study, we propose TKTO that eliminates the need for paired data, enabling a more data-efficient training paradigm, and directly targets token-level units, automatically providing fine-grained alignment signals without token-level annotations. TKTO improves the challenging Japanese TTS accuracy by 39% and reduces CER by 54%, leveraging $6\times$ more training data and assigning $12.8\times$ stronger reward to targeted tokens.

1 Introduction

Recent advances in neural network technology have enabled high-fidelity text-to-speech (TTS) synthesis models (Chen et al., 2025; Meng et al., 2025; Li et al., 2024). They typically convert input text into a phoneme sequence using a grapheme-to-phoneme (G2P) converter (Oura et al., 2010), followed by generative models from the phoneme sequence (Wang et al., 2025b; Nishimura et al., 2025). However, G2P is generally based on morphological analysis, and thus may fail to generate correct pronunciations in ambiguous languages like Japanese, where the reading and meaning of words can change depending on context (Figure 1).

G2P-free large language model (LLM)-based TTS methods (Wang et al., 2025a; Du et al., 2024) have shown great promise to generate context-aware pronunciations directly from raw text without G2P by leveraging large-scale natural language pretraining. Recent work (Zhang et al., 2025; Tian

	Ambiguous Sentence	Reading
(a)	このカレーは辛い (This curry is spicy)	"karai" (spicy)
(b)	この作業は辛い (This work is hard)	"tsurai" (hard)

Figure 1: Examples of ambiguity in Japanese. Although (a) and (b) contain the same word, its meaning and reading differ depending on the context.

et al., 2025; Zhang et al., 2024) has further applied Direct Preference Optimization (DPO) (Rafailov et al., 2023) to improve intelligibility, speaker similarity, and overall naturalness by enlarging the preference gap between pairwise samples.

Despite these advances, two fundamental challenges remain. **(i) Necessity of paired data:** DPO-based methods require paired desirable and undesirable outputs for the same utterance, but TTS systems often produce one-sided results, where many samples are consistently desirable or undesirable. This causes significant data inefficiency and wastes costly human feedback, limiting the scalability of DPO-based preference alignment. **(ii) Sample-level optimization:** Pronunciation generation is essentially a character- or token-level task, while preference alignment is conducted with utterance-level labels. This mismatch forces the model to optimize at the data-sample level rather than directly at the pronunciation unit level, leading to suboptimal learning signals and limiting the effectiveness of alignment.

To address these challenges, we propose a novel preference optimization framework, **Token-level Kahneman-Tversky Optimization (TKTO)** that constructs contrastive LLMs to estimate token-level weights and optimizes token-level preferences grounded in Kahneman-Tversky’s prospect theory (Ethayarajh et al., 2024). Our contributions are summarized in three key dimensions:

- **Problem formulation:** we pioneer the challeng-

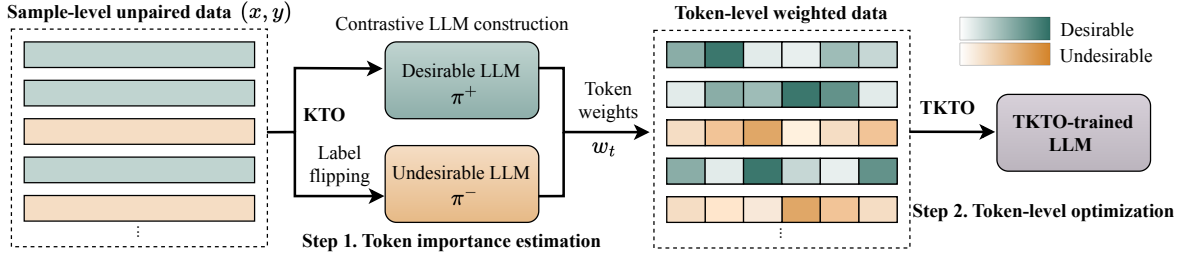


Figure 2: Overview of our TKTO framework. Step 1: we estimate token-level importance weights, constructing two contrastive LLMs. Step 2: we optimize token-level preferences.

ing task of ambiguous pronunciation as a token-level preference optimization problem.

- **Novel methodology:** we propose TKTO that (i) eliminates the need for paired data, enabling a more data-efficient training paradigm, and (ii) directly targets token-level units, automatically providing fine-grained alignment signals.
- **Wide evaluation:** we demonstrate that TKTO reduces character error rate (CER) by 54% and improves the accuracy of Japanese pronunciation by 39%. By leveraging unpaired data, TKTO can utilize $6\times$ more training data than DPO, leading to higher data efficiency. It also automatically assigns $12.8\times$ stronger rewards to targeted tokens.

2 LLM-based Text-to-Speech

We consider a text-to-speech (TTS) task formulated as conditional generation from input text x to output speech token sequence $y = (y_1, \dots, y_T)$. Let $\pi_\theta(y | x)$ denote a LLM decoder that autoregressively predicts acoustic tokens conditioned on textual input and previously generated tokens:

$$\pi_\theta(y | x) = \prod_{t=1}^T \pi_\theta(y_t | x, y_{<t}). \quad (1)$$

The output token sequence y is transformed into speech audio via a neural vocoder. Our goal is to train the LLM decoder π_θ to learn *token-level preferences* from user feedback or preference data.

3 Proposed Method

We propose a two-step approach to optimize token-level preferences from unpaired data (Figure 2). First, we quantify how informative each token is for preference learning. These weights are then used to guide our TKTO objective.

3.1 Targeted Token Weight Estimation

We first construct two contrastive language models, π^+ and π^- , to capture token-level preferences.

Then, we use the log-ratio between these two models to estimate token-level importance weights.

KTO-based Contrastive LLM Construction.

We propose a KTO-based approach to construct contrastive LLMs. KTO (Ethayarajh et al., 2024) uses unpaired data, enabling more flexible training. Details are provided in Appendix A. We build two contrastive LLMs, π^+ and π^- , which do not share parameters. We train π^+ using the original desirable/undesirable labels, and obtain π^- by training on the same data with the labels swapped, i.e., treating desirable samples as undesirable and vice versa.

Token-level Importance Sampling. We estimate the token’s weight w_t (Liu et al., 2025) as:

$$w_t = \exp(\mu \cdot \text{clamp}(\log \frac{\pi^+(y_t | x, y_{<t})}{\pi^-(y_t | x, y_{<t})}, L, U)), \quad (2)$$

where $\log \frac{\pi^+(y_t | x, y_{<t})}{\pi^-(y_t | x, y_{<t})}$ estimates the token’s reward (Rafailov et al., 2024). $L < U$ are lower and upper clamping bounds to improve optimization stability. μ is a scaling factor that controls the strength of reward weighting. For desirable samples, we set $\mu > 0$; for undesirable ones, we set $\mu < 0$.

3.2 Token-level KTO

We extend KTO (Ethayarajh et al., 2024) to the *token level*, proposing Token-level KTO (TKTO) to learn finer-grained token-level signals.

Token-level Reward and Reference. We define the reward $r_{\theta,t}(x, y)$ for each token y_t as its log-ratio against a reference policy π_{ref} :

$$r_{\theta,t}(x, y) = \log \frac{\pi_\theta(y_t | x, y_{<t})}{\pi_{\text{ref}}(y_t | x, y_{<t})}, \quad (3)$$

$$z_{0,t} = \text{KL}(\pi_\theta(\cdot | x, y_{<t}) \parallel \pi_{\text{ref}}(\cdot | x, y_{<t})), \quad (4)$$

where $z_{0,t}$ is a fixed reference baseline estimated across a microbatch (no gradients propagated).

Model	PO Data	Female			Male		
		Acc ↑	CER ↓	Bad ↓	Acc ↑	CER ↓	Bad ↓
gpt-4o-mini-tts	-	0.900	0.109	0.079	0.939	0.111	0.062
gemini-2.5-flash-preview-tts	-	0.776	0.140	0.105	0.769	0.134	0.091
gemini-2.5-pro-preview-tts	-	0.871	0.127	0.094	0.885	0.119	0.073
F5-TTS (Chen et al., 2025)	-	0.498	0.173	0.189	0.500	0.177	0.183
F5-TTS with G2P (Oura et al., 2010)	-	0.500	0.136	0.100	0.500	0.146	0.107
Base model (Du et al., 2024)	-	0.683	0.128	0.090	0.668	0.138	0.095
Supervised Fine-Tuning (SFT)	Desirable	0.674	0.119	0.076	0.654	0.130	0.084
DPO (Tian et al., 2025)	Paired	0.706	0.120	0.076	0.693	0.130	0.082
KTO (Ethayarajh et al., 2024)	Paired	0.654	<u>0.066</u>	<u>0.028</u>	0.651	<u>0.074</u>	0.030
KTO (Ethayarajh et al., 2024)	Unpaired	<u>0.933</u>	0.079	0.030	<u>0.952</u>	0.087	0.032
TKTO (ours)	Paired	0.681	0.059	0.025	0.701	0.066	<u>0.029</u>
TKTO (ours)	Unpaired	0.949	<u>0.075</u>	0.029	0.958	0.085	0.027

Table 1: Accuracy (Acc), Character Error Rate (CER), and bad case ratio (Bad) across different models. The best and second-best results are highlighted in **bold** and underline, respectively.

Token-level Value Function. We define each token’s value v_t using a logistic-shaped function:

$$v_t(x, y) = \begin{cases} \lambda_D \sigma(\beta (r_{\theta,t}(x, y) - z_{0,t})) & \text{if } y \sim y_{\text{desirable}} \mid x, \\ \lambda_U \sigma(\beta (z_{0,t} - r_{\theta,t}(x, y))) & \text{if } y \sim y_{\text{undesirable}} \mid x, \end{cases} \quad (5)$$

where $\sigma(\cdot)$ is the sigmoid function, β controls curvature, and λ_D, λ_U adjust aversion weights.

Objective Function. The TKTO loss is a weighted sum of token-level values, with importance weights w_t :

$$L_{\text{TKTO}} = \mathbb{E}_{(x,y)} \left[- \sum_{t=1}^{|y|} w_t \cdot v_t(x, y) \right]. \quad (6)$$

4 Experiments

4.1 Experimental Setup

Text Dataset. To evaluate not only CER but also more challenging cases of ambiguous Japanese pronunciation, we created a Japanese dataset of 5,000 sentences containing the word "辛い" using GPT-5 (OpenAI, 2025b). The word can be pronounced either as karai (spicy) or tsurai (hard), depending on the context; we ensured that each pronunciation appears in half of the samples.

Speech Dataset. For each text sentence, we generate five speech samples for both female and male Japanese speakers (Okamoto et al., 2023) using a TTS model, with training data containing 23 hours of speech for each speaker. Among them, the sample with the correct pronunciation and the lowest

CER is selected as a desirable sample. Conversely, the sample with the incorrect pronunciation and the highest CER is selected as an undesirable sample. The CER is computed using the whisper-v3-large (Radford et al., 2023).

Baselines. We use CosyVoice2 (0.5B) (Du et al., 2024) fine-tuned on 20K hours of Japanese speech data as our base model and apply our TKTO. The following baselines are considered:

- *Preference Optimization (PO) methods:* DPO uses 1.5K paired desirable–undesirable samples; KTO uses 9K unpaired desirable or undesirable samples; SFT uses 6K desirable samples.
- *TTS models:* Flow matching-based F5-TTS (Chen et al., 2025) trained on the 20K hours, w/ and w/o Japanese G2P (Oura et al., 2010).
- *Reference industry models:* gpt-4o-mini-tts (OpenAI, 2025a) (*Coral*: female, *Ash*: male) and gemini-2.5-tts-preview (Google, 2025) (*Zephyr*: female, *Puck*: male).

Objective Metrics: Accuracy measures the proportion of samples in which the generated pronunciation matches the correct reading for a target ambiguous word (e.g., "辛い"). We also use CER to evaluate overall robustness and the bad case ratio (Bad), where samples with CER exceed 0.3.

Subjective Metrics: We employ the naturalness mean opinion score (NMOS) to evaluate the naturalness. In addition, we conduct an ABX test. The instructions for the subjective evaluation are as follows. Three Japanese native human annotators listened to the speech samples and rated their naturalness as NMOS on a 5-point scale, where 1

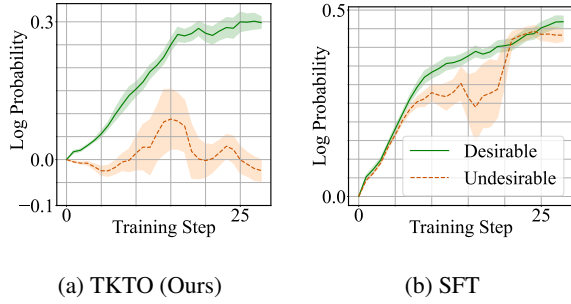


Figure 3: Average log-likelihood for desirable and undesirable tokens during training. TKTO effectively increases only that of desirable tokens.

indicates very unnatural and 5 indicates completely natural. In addition, we conducted an ABX test in which the participant listened to two generated speech samples from different models but based on the same input and then chose the one that sounds more natural; if the samples are too similar to distinguish, they are instructed to indicate a tie. The annotator was recruited via a crowdsourcing platform and received appropriate compensation for their work.

4.2 Objective Evaluation Results

Main Results on Japanese. Table 1 presents the objective evaluation results. Our TKTO model achieves the highest Japanese TTS accuracy and the lowest CER and bad case ratio. The non-LLM baseline F5-TTS shows extremely low accuracy regardless of whether it uses G2P or not, highlighting the importance of LLMs to consider contextual information. By leveraging unpaired data, we can utilize up to $6\times$ more training samples than paired methods. This leads to a significant improvement in accuracy ($0.668 \rightarrow 0.958$), surpassing even strong industry models. Paired training relies on a very narrow and biased subset that represents only rare borderline cases where both correct and incorrect pronunciations appear for the same text. This limits the learning signal and reduces generalization. Compared with KTO, our model achieves improvements across all metrics, demonstrating the effectiveness of token-level targeted optimization.

Training Dynamics. Figure 3 illustrates the changes in the average log-likelihood of desirable and undesirable tokens. As training progresses, TKTO (left) effectively increases only that of desirable tokens, whereas SFT (right) tends to increase the log-likelihood of undesirable tokens.

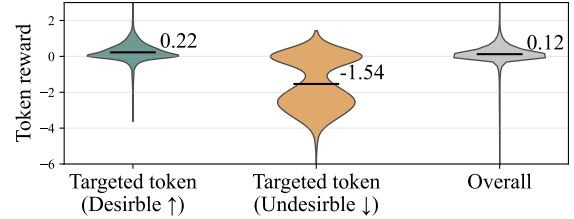


Figure 4: Token reward analysis. Desirable tokens have a higher reward, while undesirable tokens have a much lower reward, encouraging both weights to increase.

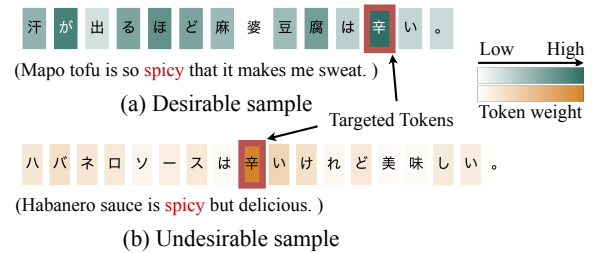


Figure 5: Case study of token weight estimation. The tokens of target character have higher weights.

Token Weight Analysis. Figure 4 illustrates the token rewards $\log \frac{\pi^+(y_t|x, y^{<t})}{\pi^-(y_t|x, y^{<t})}$ for desirable and undesirable tokens for the target character "辛" and the overall average. The desirable tokens have a higher reward (0.22) compared to the overall mean (0.12), while the undesirable tokens show a much lower value (-1.54), indicating that targeted tokens are automatically assigned larger weights. For undesirable samples, two peaks are observed when a clear difference between π^+ and π^- emerges; the reward drops sharply, whereas when no significant difference exists.

Case Study. Figure 5 presents a case study showing that the tokens corresponding to the target characters receive notably higher weights compared to non-targeted tokens.

4.3 Subjective Evaluation Results

Table 2 presents the NMOS subjective evaluation scores. Figure 6 shows the results of the ABX test. In the subjective evaluations, TKTO also outperforms the base model and the standard KTO.

4.4 Computational Cost and Data Efficiency

Building Two Contrastive LLMs. Our TKTO framework requires training two additional contrastive LLMs to estimate token weights, which introduces an extra pre-computation step in the pipeline. However, the additional training cost

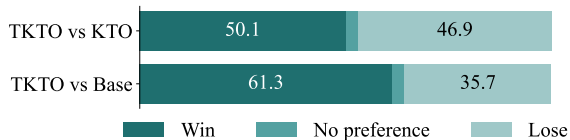


Figure 6: Results for ABX preference test.

	Base	KTO	TKTO (ours)
NMOS \uparrow	4.09	4.17	4.21

Table 2: NMOS comparison (higher is better).

is relatively small: in our setup, training these two contrastive LLMs takes approximately 10 minutes on $8 \times A100$ GPUs. This overhead is negligible compared to the overall TTS training process, where pre-training on tens of thousands of hours of data typically requires weeks of computation.

Data Efficiency with Limited Pairs. In our dataset, only 10.5% of utterances contain both a desirable and an undesirable output, while 89.5% contain only one side. This imbalance occurs because modern TTS systems tend to produce consistently correct or incorrect pronunciations, making paired samples rare. As a result, paired methods such as DPO can use only 1.5K samples, whereas unpaired methods can utilize 9K samples. In our experiments, TKTO is able to leverage the unpaired samples, resulting in approximately $6 \times$ more usable data than what is accessible to paired preference optimization methods.

4.5 Additional Validation on Chinese

In addition to Japanese, we evaluate the generalization capability of our method on Chinese. Following the same experimental setup as in the Japanese evaluation, we generate 5,000 sentences containing the polyphonic character "行" which has two context-dependent readings, "xíng" and "háng." For each sentence, we generate five speech samples using the base TTS model with a Chinese speaker, resulting in 1.5K paired samples and 4K unpaired samples. We evaluate the correctness of each reading using the wav2vec2-mms-1b-cmn-phonetic checkpoint¹.

Results. Table 3 presents the Chinese evaluation results. TKTO achieves the highest accuracy and the lowest CER and bad-case ratio among all compared methods. These results demonstrate that

¹<https://huggingface.co/Chuatury/wav2vec2-mms-1b-cmn-phonetic>

Moethods	PO Data	Acc \uparrow	CER \downarrow	Bad \downarrow
Base model	–	0.645	0.024	0.010
DPO	Paired	0.682	0.021	0.007
KTO	Paired	0.712	0.026	0.009
KTO	Unpaired	0.832	0.018	0.010
TKTO (ours)	Paired	0.746	0.024	0.007
TKTO (ours)	Unpaired	0.898	0.014	0.003

Table 3: Results on Chinese polyphonic disambiguation.

TKTO generalizes beyond Japanese and is effective for polyphonic disambiguation in Chinese.

5 Conclusion

We propose TKTO that eliminates the need for paired data, enabling a more data-efficient training paradigm, and directly targets token-level units, automatically providing fine-grained alignment signals without token-level annotations. TKTO serves as an off-policy approach compatible with human feedback, while extending it to an on-policy setting remains a promising direction for future work.

Limitations

The evaluations are conducted on only challenging Japanese and Chinese data. Although multiple preference optimization methods and configurations were tested, we only evaluate a single model size (0.5B), as larger versions of the same backbone are not publicly available and computational constraints prevent extensive pretraining and scaling experiments. Investigating scaling behavior and extending the approach to other model architectures remain to be explored. While this work only focuses on preference optimization for TTS, the proposed TKTO framework can potentially be applied to other text generation tasks where specific tokens play a crucial role.

Acknowledgments

The authors would like to sincerely thank the anonymous reviewers for their valuable comments and helpful suggestions.

Ethical Considerations

While text-to-speech (TTS) technology may raise concerns regarding unauthorized generation or misuse, the models utilized in this paper were used for research purposes under controlled conditions.

References

- Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. 2025. F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6255–6271.
- Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, and 1 others. 2024. Cosyvoice 2: Scalable streaming speech synthesis with large language models. *arXiv preprint arXiv:2412.10117*.
- Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Model alignment as prospect theoretic optimization. In *Forty-first International Conference on Machine Learning*.
- Google. 2025. Gemini 2.5-preview-tts. <https://ai.google.dev/gemini-api/docs/speech-generation>.
- Xiang Li, FanBu FanBu, Ambuj Mehrish, Yingting Li, Jiale Han, Bo Cheng, and Soujanya Poria. 2024. Cm-tts: Enhancing real time text-to-speech synthesis efficiency through weighted samplers and consistency models. In *NAACL-HLT (Findings)*.
- Aiwei Liu, Haoping Bai, Zhiyun Lu, Yanchao Sun, Xiang Kong, Xiaoming Simon Wang, Jiulong Shan, Albin Madappally Jose, Xiaojiang Liu, Lijie Wen, Philip S. Yu, and Meng Cao. 2025. TIS-DPO: Token-level importance sampling for direct preference optimization with estimated weights. In *The Thirteenth International Conference on Learning Representations*.
- Lingwei Meng, Long Zhou, Shujie Liu, Sanyuan Chen, Bing Han, Shujie Hu, Yanqing Liu, Jinyu Li, Sheng Zhao, Xixin Wu, Helen M. Meng, and Furu Wei. 2025. Autoregressive speech synthesis without vector quantization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1287–1300.
- Yuto Nishimura, Takumi Hirose, Masanari Ohi, Hideki Nakayama, and Nakamasa Inoue. 2025. Hall-e: Hierarchical neural codec language model for minute-long zero-shot text-to-speech synthesis. In *The Thirteenth International Conference on Learning Representations*.
- Takuma Okamoto, Yoshinori Shiga, and Hisashi Kawai. 2023. Hi-Fi-CAPTAIN: High-fidelity and high-capacity conversational speech synthesis corpus developed by NICT. <https://astrec.nict.go.jp/en/release/hi-fi-captain/>.
- OpenAI. 2025a. gpt-4o-mini-tts. <https://platform.openai.com/docs/models/gpt-4o-mini-tts>.
- OpenAI. 2025b. Gpt-5. <https://openai.com/gpt-5/>.
- Keiichiro Oura, Shinji Sako, and Keiichi Tokuda. 2010. Japanese text-to-speech synthesis system: Open jtalk. In *Proc. ASJ*, pages 343–344.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.
- Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. 2024. From $\$r\$$ to $\$q^*\$$: Your language model is secretly a q-function. In *First Conference on Language Modeling*.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- Jinchuan Tian, Chunlei Zhang, Jiatong Shi, Hao Zhang, Jianwei Yu, Shinji Watanabe, and Dong Yu. 2025. Preference alignment improves language model-based tts. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, and 1 others. 2025a. Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens. *arXiv preprint arXiv:2503.01710*.
- Yuancheng Wang, Haoyue Zhan, Liwei Liu, Ruihong Zeng, Haotian Guo, Jiachen Zheng, Qiang Zhang, Xueyao Zhang, Shunsi Zhang, and Zhizheng Wu. 2025b. MaskGCT: Zero-shot text-to-speech with masked generative codec transformer. In *The Thirteenth International Conference on Learning Representations*.
- Dong Zhang, Zhaowei Li, Shimin Li, Xin Zhang, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2024. Speechalign: Aligning speech generation to human preferences. *Advances in Neural Information Processing Systems*, 37:50343–50360.
- Xueyao Zhang, Yuancheng Wang, Chaoren Wang, Ziniu Li, Zhuo Chen, and Zhizheng Wu. 2025. Advancing zero-shot text-to-speech intelligibility across diverse domains via preference alignment. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12251–12270.

Appendix

A Kahneman–Tversky Optimization

Kahneman–Tversky Optimization (KTO) (Ethayarajh et al., 2024) is a pair-free alignment objective that directly maximizes a proxy of human utility using binary feedback indicating whether

an output is *desirable* or *undesirable*. Unlike DPO (Rafailov et al., 2023), KTO does not require paired comparisons.

Reference-adjusted reward. Given an input x , output y , current policy π_θ , and reference policy π_{ref} , we define the reward $r_\theta(x, y)$ as:

$$r_\theta(x, y) = \log \frac{\pi_\theta(y | x)}{\pi_{\text{ref}}(y | x)}. \quad (7)$$

A reference point z_0 is defined as:

$$z_0 = \text{KL}(\pi_\theta(\cdot | x) \| \pi_{\text{ref}}(\cdot | x)), \quad (8)$$

which captures the expected reward.

Value function. To model loss aversion and diminishing sensitivity, KTO uses a logistic value function:

$$v(x, y) = \begin{cases} \lambda_D \sigma(\beta(r_\theta(x, y) - z_0)), & \text{if } y \sim y_{\text{desirable}} | x, \\ \lambda_U \sigma(\beta(z_0 - r_\theta(x, y))), & \text{if } y \sim y_{\text{undesirable}} | x, \end{cases} \quad (9)$$

where $\sigma(\cdot)$ is the sigmoid function, β controls risk sensitivity, and λ_D, λ_U control gains and losses.

KTO objective. The KTO loss is defined as:

$$\mathcal{L}_{\text{KTO}}(\pi_\theta) = \mathbb{E}_{(x,y) \sim \mathcal{D}} [-v(x, y)], \quad (10)$$

Connection to TKTO. Our TKTO extends KTO by lifting the optimization from the sequence level to the token level for TTS. While KTO assigns a single utility to an entire output y , TKTO introduces token-wise weights that estimate the relative contribution of each token to human utility. By explicitly modeling heterogeneous token importance, TKTO enables fine-grained credit assignment across tokens, allowing the model to emphasize critical reasoning or decision-making steps while down-weighting less informative tokens. In this sense, TKTO can be viewed as a token-wise generalization of KTO that preserves its prospect-theoretic inductive bias while substantially improving optimization granularity.

B Implementation Details

Our implementation was based on the publicly available codes of prior work (Chen et al., 2025;

Parameter L, U	Female			Male		
	Acc \uparrow	CER \downarrow	Bad \downarrow	Acc \uparrow	CER \downarrow	Bad \downarrow
-1, 1	0.937	0.076	0.028	0.951	0.088	0.031
-2, 2	0.949	0.075	0.029	0.958	0.085	0.027
-3, 3	0.956	0.072	0.027	0.948	0.088	0.028

Table 4: Parameter sensitivity analysis.

Model	Female		Male	
	CER \downarrow	Bad \downarrow	CER \downarrow	Bad \downarrow
Base model	0.152	0.110	0.151	0.094
KTO	0.072	0.050	0.076	0.047
TKTO (ours)	0.065	0.044	0.069	0.045

Table 5: Evaluation results using the nvidia/parakeet-tdt_ctc-0.6b-ja ASR model.

Du et al., 2024), which are released under research-permissive licenses, and hyperparameters and libraries used followed those studies. Base model and baseline F5-TTS were initialized from the pre-trained checkpoints and fine-tuned once. Training took a few minutes on $8 \times A100$ GPUs. All preference optimization experiments were conducted on 1 epoch with $1e-6$ learning rate. We set $\lambda_D = \lambda_U = 1$, $\beta = 0.10$, we set $\mu = 1$ for desirable tokens, $\mu = -1$ for undesirable ones, following Ethayarajh et al. (2024).

C Parameter Sensitivity Analysis

Since the clamping range controls the scaling of the reward weights, it plays an important role in balancing learning stability and sensitivity. Table 4 presents the clamping range (L,U) sensitivity analysis results. While performance remains stable for moderate clamping ranges, wide bounds (e.g., (-3, 3)) can slightly lead to accuracy degradation. This suggests that imposing reasonable constraints on the reward range is beneficial for achieving optimal results. We select $(-2, 2)$ as the optimal range for experiments.

D Robustness across ASR Backends

To verify the robustness of our evaluation, we conduct additional experiments using the parakeet-tdt_ctc-0.6b-ja² ASR model with paired preference data. Table 5 shows consistent improvements over the baselines, mirroring the trend observed with Whisper. As shown in Table 5, TKTO achieves the lowest CER and Bad ratios

²https://huggingface.co/nvidia/parakeet-tdt_ctc-0.6b-ja

for both speakers, confirming that the performance gains are not tied to a particular ASR backend.