SALT: Speech and Language Transformer

Ksenia Sycheva
Vikhr Models
KS9270022@gmail.com

Aleksandr Nikolich
Vikhr Models
alexwortega@yandex.ru

Konstantin Korolev
HSE University
korolevko@icloud.com

Ilya Kuleshov
HSE University
kul757.48@mail.ru

Sergey Bratchikov
Vikhr Models
hivaze.me@gmail.com

Igor Kiselev
Accenture
igor.kiselev@accenture.com

Andrey Kuznetsov
AIRI, Innopolis University
kuznetsov@airi.net

Abstract

Recent work has seen growing interest in multimodal AI systems, yet many existing approaches rely on complex pipelines with disjointed components for text and audio processing. Meanwhile, few such systems are optimized for Slavic languages, which present unique linguistic challenges. We introduce SALT, a novel end-to-end transformer architecture for neural speech synthesis that unifies text and audio processing within a single framework. By employing a unified tokenization strategy, SALT seamlessly integrates text and audio tokens, enabling high-quality, real-time inference on modern GPUs. Our approach streamlines the traditional text-to-speech pipeline by removing the need for intermediate representations and specialized models, resulting in a simpler multimodal system. Our experiments demonstrate SALT's competitive performance in multilingual speech synthesis, including Slavic languages.

1 Introduction

Large language models (LLMs) like ChatGPT have become indispensable tools for professionals. Recently, adaptations of these models to various modalities such as speech were developed (OpenAI, 2024), which make their usage more intuitive and accessible for more users. Existing approaches that combine speech and text modalities typically rely on separate adapters or convert text into phonetic representation, and as a result, suffer from high latency due to fragmented pipelines (Zhang et al., 2023a; Chu et al., 2023; Palaskar et al., 2024). Prior works have demonstrated the potential of unified multimodal architectures for real-time interaction, but such systems remain English-centric: many of these works have largely overlooked Russian and other Slavic languages, where phoneme-grapheme relationships can be more complex, while high-quality training data is limited compared to English.

To address this gap, we present **Speech And Language Transformer (SALT)**, a single-model architecture for low-latency multilingual speech interaction, which eliminates pipeline latency by processing speech/text in the shared latent space. SALT extends a pretrained LLM with audio tokens, enabling joint ASR/TTS via a unified loss function. It requires about 150 hours on Nvidia H100 for training to achieve reasonable performance on speech datasets. Additionally, we release a collection of Slavic audio datasets[1]. By training SALT on Slavic datasets, we demonstrate its suitability for Slavic languages, achieving competitive results in speech synthesis while maintaining low latency. SALT checkpoints[2] and training code[3] are open-sourced.

In the remaining paper, we discuss existing approaches targeting audio and text modalities. In Section 3 we compare various audio tokenization methods and describe our training procedure. Lastly, we detail experimental setup and analyze results in Section 4.

2 Related Work

While numerous studies have investigated speech-to-text and text-to-speech multimodal models, two primary approaches have emerged, which we outline below.

First approach utilizes adapters to encode new modalities (Liu et al., 2023; Zhang et al., 2023a;

[1] https://huggingface.co/collections/Vikhrmodels/tone-6846b76d036abfe1c54f0a89

[2] https://huggingface.co/collections/Vikhrmodels/salt-6869033876b0f112c0e6da22

[3] https://github.com/VikhrModels/Salt

Chu et al., 2023). This approach has demonstrated success in adapting LLMs to new input modalities, enabling models like VITA (Fu et al., 2024) to process text, images, audio, and video within a unified framework. While it leverages pre-trained LLMs without full retraining, it relies on auxiliary models (e.g., TTS for speech synthesis or diffusion models for image generation) to produce non-textual outputs. A key limitation, however, is information loss during cross-modal conversion, as the system must translate between discrete representations. Despite this, the approach remains practical and scalable, benefiting from the strong foundational capabilities of existing language models.

The second approach involves training models from scratch using autoregressive objectives, as exemplified by architectures like VALLE (Wang et al., 2023) and FishSpeech (Liao et al., 2024). These systems unify text and audio modalities by first discretizing speech signals into a sequence of tokens using VQ-VAE-based audio tokenizers (van den Oord et al., 2017). This method preserves high-fidelity audio representations while enabling seamless integration with text tokens in a shared latent space. However, they require full training from scratch, preventing leverage of pre-existing LLM knowledge.

In our work, we extend existing pre-trained text-only LLMs to a new modality by incorporating discrete audio tokens. This enables the model to retain some of its natural language understanding capabilities while also gaining the benefits of the second approach – namely, high-fidelity audio modeling through tokenized representations in a unified sequence space.

3 Method

In contrast to prior works, SALT combines the strengths of adapter-based adaptation and end-to-end autoregressive training by integrating:

1. Unified speech and text representation: by processing both speech and text as discrete tokens, SALT eliminates the need for modality-specific encoders or cross-modal conversion.

2. A pre-trained LLM: instead of training from scratch, backbone is initialized from pre-trained LLM.

SALT architecture is illustrated in Figure 1. Below we describe audio tokenization and training procedure in more details.

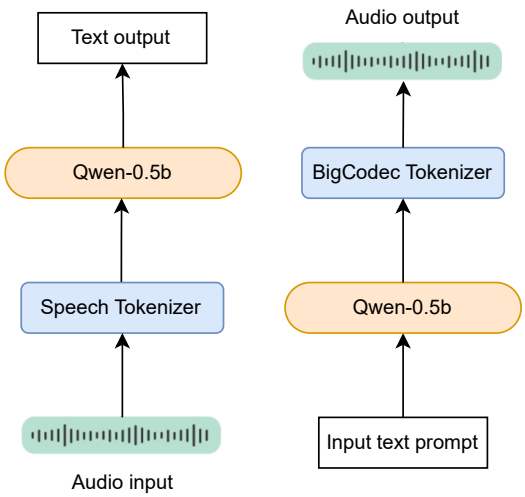

Figure 1: Architecture of SALT showing the unified processing pipeline for both ASR and TTS tasks. The model extends a pre-trained LLM with parallel audio token streams (SpeechTokenizer for ASR, BigCodec for TTS), enabling joint speech-text understanding and generation through a shared latent space.

3.1 Speech Tokenization

To achieve seamless integration of audio with text representations, we adopted a tokenization approach using pre-trained speech tokenizers. This enables audio signals to be processed at the same semantic level as text tokens within our unified architecture. We experimented with three state-of-the-art tokenizers, which are discussed below.

SpeechTokenizer (Zhang et al., 2023b) employs a hierarchical speech representation that explicitly separates semantic content (encoded in the first quantizer) from acoustic details (preserved in subsequent quantizers). During inference, arbitrary number of quantizers can be used. Such features disentanglement proved to be important for training SALT on ASR.

WavTokenizer (Ji et al., 2025) compresses the quantizer layers of acoustic codec models to a single quantizer. It achieves state-of-the-art reconstruction quality, while using fewer tokens (75 tokens per second compared to 300 tokens per second used by SpeechTokenizer). WavTokenizer is useful in TTS, where it enabled high-fidelity audio synthesis with reduced computational overhead.

BigCodec (Xin et al., 2024) employs a novel single-quantizer architecture operating with an extended codebook (8,192 entries), enabling

low-bitrate speech coding without hierarchical quantization. BigCodec gives better reconstruction quality on Russian and other Slavic languages.

SALT supports flexible tokenizer choices – different tokenizers could be used for ASR and TTS, further enhancing its adaptability to diverse tasks. In the main experiments we used two different tokenization methods – BigCodec was adopted to generate audio in text-to-speech tasks, while SpeechTokenizer was utilized for speech recognition. We additionally train SALT variant with WavTokenizer, which demonstrated high quality on text-to-speech task; however, its effectiveness is limited in Slavic languages. Further we use $\mathcal{V}_{\text{asr}}, \mathcal{V}_{\text{tts}}$ for set of codebooks used in speech recognition and speech synthesis, respectively.

3.2 Training Methodology

In contrast to (Liao et al., 2024), SALT is trained in an end-to-end fashion. Its backbone is initialized from a pre-trained LLM. Audio inputs are first tokenized using a neural audio codec that was chosen for a task. LLM vocabulary $\mathcal{V}_{\text{lm}}$ is extended with discrete tokens from neural codecs, thus eventual size of vocabulary is

$$|\mathcal{V}_{\text{total}}| = |\mathcal{V}_{\text{lm}}| + |\mathcal{V}_{\text{asr}}| + |\mathcal{V}_{\text{tts}}| \quad (1)$$

In this setup text and audio representations are treated on the same level. This allows the model to jointly process interleaved sequences of speech and text tokens within a single autoregressive framework. To align text and audio modalities we finetune base model on ASR and TTS sequences with the following prompt templates:

ASR Template: `<|start_of_sequence|>`
`text tokens <|start_of_audio|>`
`audio tokens <|end_of_audio|>`
`<|end_of_sequence|>`

TTS Template: `<|start_of_sequence|>`
`<|start_of_audio|> audio tokens`
`<|end_of_audio|> text tokens`
`<|end_of_sequence|>`

3.3 Tone Datasets

As discussed above, there is significantly less audio data available in Slavic languages compared to English. To address this gap, we release a collection of Slavic-language audio datasets, totaling over 4,000 hours. Below, we detail the preparation of these datasets. Descriptive statistics are reported in Table 1.

ToneSlavic is a subset of multilingual Common Voice 2.1 (Ardila et al., 2020) that consists of Russian, Ukrainian, and Belarus languages only.
ToneWebinars - a processed version of ZeroAgency/shkolkovo-bobr.video-webinars-audio[4], which includes audio from webinars.
ToneSpeak is a large-scale Russian-language audio dataset with detailed annotations for intonation, timbre, and emotional vocal characteristics. Data generation was done in two stages. First, text prompts were generated using GPT-4.1-mini (OpenAI et al., 2024). Then these prompts were used to synthesize speech with GPT-4o-mini (OpenAI, 2024) using ten different voices.
ToneBooks contains Russian audiobooks available online. We manually removed advertisements, footnote references, and other non-speech content, followed by forced alignment using EchoGarden[5].

4 Experiments

Our experimental evaluation assesses SALT's capabilities across both speech recognition and synthesis tasks. We establish competitive baselines, evaluate on standardized benchmarks, and analyze performance through: (1) *quantitative metrics* comparing against state-of-the-art systems, (2) *modality-specific analyses* for ASR and TTS. All experiments use fixed random seeds to ensure reproducibility. Generation arguments used for inference are listed in Appendix A.2.

4.1 Experimental Setup

Datasets For speech recognition (ASR) evaluation, we use the LibriSpeech *test-clean* split (Panayotov et al., 2015) as the primary English benchmark. Text-to-speech (TTS) quality is evaluated on the Mozilla Common Voice Russian test set (Ardila et al., 2019), with additional cross-lingual analysis conducted on LibriSpeech *test-clean* to verify model generalization. All audio was resampled to 16 kHz prior to evaluation. For training we combine English LibriSpeech and ToneSlavic.

Models. We systematically evaluate three configurations, specifically testing whether multitasking degrades performance:

[4] https://huggingface.co/datasets/ZeroAgency/shkolkovo-bobr.video-webinars-audio
[5] https://github.com/echogarden-project/echogarden

Table 1: Statistics of tone datasets, showing file counts, total duration, and utterance lengths for train/validation splits. Note the diversity in scale: ToneSlavic has the largest file count (1.4M train samples), while ToneWebinars contains the longest utterances (median 28s). ToneBooks and ToneSlavic feature shorter samples (median 5.6s and 4.5s, respectively).

Dataset	Split	Files	Duration (hrs)	Avg Length (s)	Median Length (s)
ToneSpeak	Train	6,298	26.33	15.05	14.86
	Valid	700	2.91	14.95	14.81
ToneBooks	Train	91,976	179.16	7.01	5.62
	Valid	4,841	9.42	7.00	5.63
ToneSlavic	Train	1,475,164	1,964.71	4.79	4.50
	Valid	1,265,35	168.62	4.80	4.50
ToneWebinars	Train	286,787	2053.55	25.78	28.00
	Valid	21,587	154.34	25.74	28.00

- **SALT (ASR)**: Specialized for speech recognition (English only).
- **SALT (TTS)**: Optimized for text-to-speech synthesis (English, Russian, and Ukrainian).
- **SALT Unified (ASR + TTS)**: Unified model handling both tasks.

For encoding audio in speech recognition, we use SpeechTokenizer (Zhang et al., 2023b), due to its ability to separate semantic information from acoustic. For speech generation BigCodec (Xin et al., 2024) is used, because of high-fidelity reconstruction of audio in Russian and Ukrainian languages. Qwen-2.5 with 0.5b parameters (Yang et al., 2024) is used in all experiments as SALT backbone.

Training Infrastructure

Training was conducted on 1× H100 GPU using PyTorch and accelerate for approximately 168 GPU hours.

Baselines

We evaluate SALT against state-of-the-art systems in both automatic speech recognition (ASR) and text-to-speech (TTS) synthesis. Our comparisons include recent open-source models and published results on standard benchmarks. For ASR evaluation, we compare against **Ichigo** [6], an end-to-end multilingual ASR system. For TTS comparisons, we use **Fish Audio** (Liao et al., 2024) as our primary baseline, representing state-of-the-art in multilingual expressive TTS with voice cloning capabilities.

[6] https://menlo.ai/blog/llama3-just-got-ears

Metrics

ASR. To evaluate SALT on speech recognition, we employ Word Error Rate (WER), which measures transcription fidelity. To evaluate more fine-grained character/phoneme level accuracy, we report Character Error Rate (CER).

TTS. For evaluating speech generation we compute three standard audio quality metrics:

- Perceptual Evaluation of Speech Quality (PESQ): correlates with human judgments of naturalness.
- Short-Time Objective Intelligibility (STOI): intelligibility prediction.
- Scale-Invariant Signal-to-Distortion Ratio (SI-SDR): measures waveform preservation.

4.2 Results

Performance on speech recognition. Table 2 compares ASR performance on LibriSpeech test-clean. The unified SALT model achieves competitive results in terms of Ichigo WER score, while both SALT variants show higher throughput. Unified *SALT (ASR+TTS)* outperforms the *SALT (ASR only)* variant (7.42% vs 8.42% CER), suggesting joint training may provide regularization benefits for speech recognition.

Performance on text-to-speech. Table 3 compares TTS quality on the subset of Mozilla Russian and Ukrainian test split. The unified SALT (ASR+TTS) achieves competitive PESQ and STOI, closely matching Fish Audio, while requiring significantly less compute (168 H100 hours vs Fish

Table 2: ASR Performance (LibriSpeech test-clean)

Model	CER ↓	WER ↓	Chars/s ↑
Ichigo 3.1B	**4.75**	**16.42**	118.2
SALT (ASR only)	8.42	18.49	**191.26**
SALT (ASR+TTS)	7.42	16.49	179.7

Audio's 1344 H100 hours + 1344 RTX 4090 hours). SALT (TTS-only) achieves superior SI-SDR (24.84 dB vs joint training's 23.09 dB), while joint training preserves intelligibility (STOI=0.18) despite PESQ trade-offs. The 1.8 dB SI-SDR gap between SALT variants suggests task specialization benefits audio reconstruction, while joint training offers balanced accuracy.

Ablation of audio tokenizers. In addition to the main results using BigCodec for TTS and SpeechTokenizers for ASR, we conducted supplementary experiments with WavTokenizers. This model demonstrated strong performance in synthesizing high-quality, diverse speech and zero-shot voice cloning. However, this model was less effective on ASR and Slavic languages speech synthesis. For quantitative evaluation, we computed SIM-O (objective similarity) alongside standard audio quality measures. Results are summarized in Table 4.

5 Conclusion

We introduce SALT, a unified transformer architecture for joint text and audio processing that eliminates modality-specific encoders by representing both speech and text as discrete tokens within a single model. We additionally release a large-scale collection of Slavic audio datasets. When trained on Russian data, SALT demonstrates strong cross-lingual generalization, with our unified ASR+TTS model achieving competitive STOI (0.18) and SI-SDR (23.09 dB) scores on the Mozilla Russian test set—performance comparable to specialized baselines like Fish-audio, while retaining the flexibility of joint speech-text processing.

6 Limitations

This study has several limitations. While SALT demonstrates competitive performance in Slavic-language speech generation (TTS), its automatic speech recognition (ASR) capabilities are currently limited to English. This constraint arises from the use of SpeechTokenizer for ASR, which lacks support for Slavic languages. Extending the method to Slavic ASR would require additional training of the tokenizer. Addressing this gap could further enhance the model's versatility and applicability across all target modalities and languages.

7 Acknowledgements

Innopolis University authors were supported by the Research Center of the Artificial Intelligence Institute at Innopolis University. Financial support was provided by the Ministry of Economic Development of the Russian Federation (No. 25-139-66879-1-0003).

References

R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 4211–4215.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *CoRR*, abs/1912.06670.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *Preprint*, arXiv:2311.07919.

Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Meng Zhao, Yifan Zhang, Xiong Wang, Di Yin, Long Ma, Xiawu Zheng, Ran He, Rongrong Ji, Yunsheng Wu, Caifeng Shan, and Xing Sun. 2024. Vita: Towards open-source interactive omni multimodal llm. *arXiv preprint arXiv:2408.05211*.

Shengpeng Ji, Ziyue Jiang, Wen Wang, Yifu Chen, Minghui Fang, Jialong Zuo, Qian Yang, Xize Cheng, Zehan Wang, Ruiqi Li, Ziang Zhang, Xiaoda Yang, Rongjie Huang, Yidi Jiang, Qian Chen, Siqi Zheng, and Zhou Zhao. 2025. Wavtokenizer: an efficient acoustic discrete codec tokenizer for audio language modeling. *Preprint*, arXiv:2408.16532.

Shijia Liao, Yuxuan Wang, Tianyu Li, Yifan Cheng, Ruoyi Zhang, Rongzhi Zhou, and Yijin Xing. 2024. Fish-speech: Leveraging large language models for advanced multilingual text-to-speech synthesis. *Preprint*, arXiv:2411.01156.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

OpenAI. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276. ArXiv:2410.21276 [cs.CL].

Table 3: TTS Performance Comparison (Russian/Ukrainian). **Bold** indicates best result per column.

Model	Russian			Ukrainian		
	PESQ ↑	STOI ↑	SI-SDR ↑	PESQ ↑	STOI ↑	SI-SDR ↑
Fish-audio	**1.20**	0.16	23.00	**1.13**	**0.18**	21.45
SALT (TTS)	1.11	0.16	**23.58**	1.03	0.14	**22.75**
SALT (ASR+TTS)	1.09	**0.18**	23.09	1.03	0.14	21.86

Table 4: Ablation study with a different audio tokenizer. Speech generation quality metrics for WavTokenizers (English)

Model	PESQ↑	STOI↑	SI-SDR↑	SIM-O↑
Original	4.15	0.997	27.45	-
Fish-speech	1.26	0.17	25	0.91
SALT (Wav)	1.27	0.16	20.3	0.88

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Shruti Palaskar, Oggi Rudovic, Sameer Dharur, Florian Pesce, Gautam Krishna, Aswin Sivaraman, Jack Berkowitz, Ahmed Hussen Abdelaziz, Saurabh Adya, and Ahmed Tewfik. 2024. Multimodal large language models with fusion low rank adaptation for device directed speech detection. *Preprint*, arXiv:2406.09617.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2017. Neural discrete representation learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 6309–6318.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. 2023. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*.

Detai Xin, Xu Tan, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2024. Bigcodec: Pushing the limits of low-bitrate neural speech codec. *Preprint*, arXiv:2409.05377.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, et al. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023a. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *Preprint*, arXiv:2305.11000.

Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. 2023b. Speechtokenizer: Unified speech tokenizer for speech large language models. *arXiv preprint arXiv:2308.16692*.

A Hyperparameters

A.1 Training

Hyperparameter	Value
Batch size	4
Learning rate	1e-4
Learning rate scheduler type	cosine
Warmup steps	1,000
Dropout rate	0.0
Gradient clipping	0.25
Training epochs	10
Optimizer	AdamW
Weight decay	0.1
Maximum sequence length	4096

Table 5: Training hyperparameters for SALT

A.2 Inference

For inference, random seeds 42, 43, 44 were used. Speech recognition used more deterministic settings (Table 6) for stable transcriptions, while speech generation employed higher randomness (Table 7) to produce diverse and natural-sounding outputs.

Hyperparameter	Value
Top k	20
Top p	0.99
Temperature	0.2

Table 6: Speech recognition inference hyperparameters

Hyperparameter	Value
Top k	200
Temperature	1.2

Table 7: Speech generation inference hyperparameters