

F5-TTS Fine-tuning for Kazakh Language

by

Nurbek Moldakhmetov

Submitted to the Department of Data Science
in partial fulfillment of the requirements for the degree of

Master of Science in Data Science

at the

NAZARBAYEV UNIVERSITY

June 2026

© Nazarbayev University 2026. All rights reserved.

Author

Department of Data Science

26.04.2026

Certified by

Michael Lewis

Associate Professor, School of Engineering and Digital Sciences

Thesis Course Coordinator

Certified by

Zohaib Latif

Assistant Professor, School of Engineering and Digital Sciences

Thesis Course Coordinator

Accepted by

Elizabeth Arkhangelsky

Dean, School of Engineering and Digital Sciences

F5-TTS Fine-tuning for Kazakh Language

by

Nurbek Moldakhmetov

Submitted to the Department of Data Science
on 26.04.2026, in partial fulfillment of the
requirements for the degree of
Master of Science in Data Science

Abstract

This thesis presents a cross-lingual transfer learning methodology to adapt the F5-TTS architecture for the Kazakh language, enabling zero-shot voice cloning capabilities in a low-resource setting. While diffusion transformers and flow matching based architectures are proven to have superior zero-shot capability, the requirement for massive amounts of training data (typically more than 50K hours) practically rules out their application in low-resource languages like Kazakh. Beyond this problem, there are also several specific language features of Kazakh (agglutinative morphology, strict vowel harmony and the Cyrillic script) that present challenges for pre-trained models based on English.

To bridge this technology gap, this study proposes a cross-lingual transfer learning approach. The methodology proposes a tokenizer embedding matrix surgery to cater to unique Cyrillic symbols and preserve intrinsic zero-shot intelligence. Then the research employs a multi-stage curriculum learning with 900 hours of audio, including linguistic alignment, acoustic refinement and expressive fine-tuning.

The fine-tuned model resulted in an 8.72% Word Error Rate (WER) and an objective speaker similarity (SIM-o) of 0.6073, indicating intelligibility and preservation of zero-shot voice cloning. The subjective results also show that the model is effective, with a naturalness Mean Opinion Score (MOS) of 3.75 ± 0.41 and quality MOS of 3.92/5. In summary, this experiment demonstrates the viability of targeted tokenizer surgery and curriculum learning, enabling reasonably high-quality and intelligible speech synthesis for low-resource languages, without the need for large data sets.

Thesis Supervisor: Michael Lewis

Title: Associate Professor, School of Engineering and Digital Sciences

Thesis Supervisor: Zohaib Latif

Title: Assistant Professor, School of Engineering and Digital Sciences

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Prof. Lisa Chalaguine, for accepting me as her student and shaping the direction of this research. Her guidance has been invaluable.

I am equally thankful to Prof. Siamac Fazli for his support from the very inception of this project. He was the first to recognize the potential of this topic, validating it as a strong thesis idea and providing crucial early insights. His continuous encouragement and thoughtful comments significantly improved the quality of this work.

Contents

1	Introduction	9
1.1	Motivation	9
1.2	Research Questions and Hypothesis	10
2	Related work	13
2.1	Architectural Evolution - Autoregressive vs. Non-Autoregressive	13
2.2	The Low-Resource Bottleneck	14
2.3	Cross-Lingual Transfer and Tokenization	15
2.4	Mitigating Data Scarcity via Curriculum Learning	16
2.5	Research Gap	17
3	Methodology	19
3.1	Overview	19
3.2	Computational Environment and Infrastructure	19
3.2.1	Hardware Specifications: The NVIDIA DGX-2 System	19
3.2.2	Distributed Training Strategy	20
3.3	Theoretical Framework: F5-TTS Architecture	21
3.3.1	Conditional Flow Matching (CFM)	21
3.3.2	Diffusion Transformer (DiT) Backbone	22
3.4	Data Engineering and Tokenizer Adaptation	23
3.4.1	Dataset Curation	23
3.4.2	Preprocessing Pipeline	24
3.4.3	Tokenizer Extension and Adaptation	25
3.5	Methodology: Curriculum and Transfer Learning	26
3.5.1	Stage 1: Linguistic Alignment (OpenSLR)	26

3.5.2	Stage 2: Acoustic Refinement (KazTTS2)	27
3.5.3	Stage 3: Expressive Fine-Tuning (KazEmoTTS)	27
3.6	Implementation Details and Hyperparameters	27
3.7	Evaluation Framework	28
3.7.1	Objective Metrics	28
3.7.2	Subjective Evaluation: The Telegram Bot Platform	30
4	Results	33
4.1	Training Dynamics	33
4.2	Objective Evaluation	36
4.2.1	Baseline Performance	36
4.2.2	Kazakh Adaptation Results	36
4.2.3	Analysis and Comparison	37
4.3	Subjective Evaluation: Telegram Bot Platform	39
4.3.1	Summary of Overall Performance	40
4.4	Empirical Observations on Inference Constraints	40
5	Conclusion	43
5.1	Key Contributions	43
5.2	Limitations	43
5.3	Conclusion	44

Chapter 1

Introduction

Text-to-speech (TTS) models have gone through another fundamental paradigm shift. It began as rigid rule based approaches to expressiveness using highly advanced and complex neural networks[1]. DiTs and FM are primarily responsible for the current highly expressive and generative nature of contemporary architectures like F5-TTS, which are capable of generating high-fidelity, natural-sounding speech while maintaining robust zero-shot voice cloning capabilities.[2]. These highly successful foundational models are however totally reliant upon large amounts of training data; specifically, between 50,000 and 100,000 hours of very diverse audio.[2]. Given the fact that many low resource languages contain less than 1,000 hours of high quality speech, this amount of data represents a significant barrier for low resource languages.[3].

The purpose of this paper is to develop a methodological approach for adapting the state of the art F5-TTS framework to be used with the Kazakh language. In order to address the problem of limited data availability, the proposed solution utilizes both tokenizer matrix surgery to allow for the use of a Cyrillic script and a multi stage curriculum learning pipeline. Using these two techniques will enable the development of intelligible and reasonably high-quality Kazakh speech without having to rely on extremely large datasets.

1.1 Motivation

The main reason behind conducting this research is to bridge the technology gap that exists between high resource foundation models and the reality of the Kazakh language.

One of the reasons why Kazakh is so difficult to model linguistically is because of its highly agglutinative morphology, strict vowel harmony rules, and reliance on a Cyrillic script that includes unique phonetic characters[3]. Foundation models that were developed using English as their base language do not include these characteristics in their internal representations and therefore cannot automatically handle them when they encounter them.

Additionally, due to the relatively small amount of Kazakh audio currently available, training a large Flow Matching architecture from scratch using this data results in catastrophic model collapse, severe overfitting and degradation of acoustic output[3]. Therefore, there is a strong need for developing a cross lingually efficient adaptation technique. Through the utilization of transfer learning to maintain the original pre-trained model’s ability to exhibit zero shot intelligence while extending its tokenizer through surgical modifications and progressively refining its acoustic alignment, this research hopes to provide universal access to real time, reasonably high-quality speech synthesis for all Kazakh speakers.

1.2 Research Questions and Hypothesis

To close the technological gap between the high-resource foundation models and the specific linguistic needs of the Kazakh language, this study addresses the following research questions:

- **RQ1:** How can non-autoregressive, continuous flow matching architectures (specifically the F5-TTS model) be adapted for a morphologically rich, low-resource language utilizing a non-Latin script without inducing catastrophic forgetting of the model’s pre-trained latent representations?
- **RQ2:** Does a multi-stage curriculum learning framework mitigate the negative impacts of data scarcity and varying acoustic quality when fine-tuning a large-scale zero-shot text-to-speech foundation model?

The hypothesis is that through the combination of strategic tokenizer embedding matrix surgery and a progressively structured multi-staged curriculum learning process, it should be feasible to adapt an English pre-trained F5-TTS Foundation Model to the

Kazakh language. Through this combined methodical approach, the model will be able to handle the unique Cyrillic character set as well as generate natural-sounding reasonably high-quality synthetic speech, even though only around 900 hours of training data is available; thereby enabling the model to reduce the data requirement substantially through transfer learning.

Chapter 2

Related work

The development of Text-to-speech (TTS) synthesis technology has undergone an important paradigm shift in the last years. Traditionally, TTS systems were based mainly on rule-based waveform coding, unit concatenation, and statistical parametric speech synthesis (SPSS), using hidden Markov Models (HMM)[1][4]. Although SPSS models allowed a flexible manipulation of style and emotion, their output was often characterized by over-smoothing artifacts and lacked the high fidelity acoustics detail typical of human speech[5]. Deep neural networks led to a new paradigm for TTS, based on sequence-to-sequence architecture, that maps linguistic characteristics directly to intermediate representations, such as Mel-spectrograms[5][6]. Recently, due to advances in generative modeling, most of the current TTS research focuses on large scale neural architectures - such as speech large language models (Speech-LLMs) and diffusion-based frameworks - capable of producing highly natural and expressive speech in zero-shot scenarios[7][8]. Nevertheless, the extraordinary amounts of data necessary for training these modern architectures represent a significant barrier when trying to extend their capabilities to low-resource languages. Therefore, the main focus of this review will be related to overcoming this data requirement barrier when developing low-resource languages[5][4].

2.1 Architectural Evolution - Autoregressive vs. Non-Autoregressive

Autoregressive models (AR), such as VALL-E and AudioLM, consider speech synthesis as a conditional language modeling problem where each discrete audio token is produced

sequentially by a neural codec (for example, RVQ, SNAC) [8][9][10]. These Speech-LLMs perform very well in terms of in-context learning and prosody reproduction; nevertheless, both models present some inherent limitations. [11]. The discrete nature of the tokens usually leads to lower quality reconstruction of the audio compared to continuous acoustic features [8]. Moreover, because sequential prediction of long discrete speech codes is computationally expensive and scales linearly with the length of the generated audio, the decoding process is susceptible to error accumulation during decoding[12].

On the other hand, Non-Autoregressive (NAR) models — including architectures like FastSpeech, Glow-TTS, and VITS — generate Mel-spectrograms or raw waveform data simultaneously, which reduces latency of inference by a significant amount[13][14]. While earlier NAR frameworks had problems with duration modelling and required complex alignments between phonemes, recent advances in continuous latent diffusion and Flow Matching (FM) have completely re-defined the landscape of NAR models[8][2][15]. Specifically, some models such as Voicebox, E2-TTS, and F5-TTS utilize continuous Mel-spectrograms and rely on Ordinary Differential Equations (ODEs) guided by FM to simplify training objectives and stabilize generation when compared to traditional diffusion[2][16]. Specifically, F5-TTS employs a Diffusion Transformer (DiT) backbone to execute text-guided speech infilling, avoiding complex phonetic alignment designs by predicting dynamic-length spans at fixed time steps[2][8]. Implicit alignment preserves natural prosodic flows and achieves constant-time step inference latencies making it significantly better than AR models for real-time deployments[12].

2.2 The Low-Resource Bottleneck

The extremely successful zero-shot performance of modern FM and AR foundation models are closely related to their extremely large training corpora. Generally speaking, most generative models require between 50,000 hours and over 100,000 hours of multi-domain speech data to successfully converge and map complex Text-to-speech alignments[7][8]. When working with small-scale datasets (typically under 1,000 hours; and often less than 50 hours for specialized targets), these architectures experience severe performance degradation[8][17]. According to experimental results, scaling down the dataset leads to over-fitting of pronunciation, disjointed prosody and significantly increased Word Error

Rate (WER) values[8][18].

These huge data requirements create a serious barrier for low resource languages like Kazakh[3][19]. First of all Kazakh is highly agglutinative language that has rich morphology and strict vowel harmony rules. Properties that standard models optimized for Indo-European phonologies struggle to capture[19]. While there exist high-quality open source datasets such as KazakhTTS2 with total volume typically plating at few hundred hours[3][19], they usually do not possess enough volume to train large-scale architectures from scratch without catastrophic model collapse. Adaptation of existing high-resource foundation models to low-resource languages is an operational requirement[20][18].

2.3 Cross-Lingual Transfer and Tokenization

Cross-lingual transfer learning reduces the impact of data sparsity through leveraging a model that was pre-trained on a rich resource language (such as English) and then applying it to a sparse resource target language[18][21]. One key problem with this process of transferring the knowledge from one domain to another, especially when the target language uses a non-Latin script (as is the case with Kazakh’s Cyrillic alphabet)[22][19] is the differences between the input spaces. Some methods have been developed to solve this problem by converting the target transcription to a common representation that can be utilized across all languages, such as the International Phonetic Alphabet (IPA)[21][23]; however, newer implicit alignment architectures like F5-TTS use raw character or Byte-Pair Encoding (BPE) inputs[2].

Directly integrating a new script requires vocabulary expansion, which introduces a significant risk of catastrophic forgetting within the pre-trained model’s latent space. Changing the input embedding matrix to include new Cyrillic tokens for out-of-vocabulary (OOV) words generally causes "catastrophic forgetting", where the model loses the ability to reason about the structure of the text and perform zero-shot synthesis[24][25]. Modifying an embedded representation of text changes how the text is represented and the weighting of various parts of the text. The target application will generally have a data distribution that differs dramatically from that used during model training. Therefore, Low-Rank Adaptation (LoRA) and other parameter-efficient fine-tuning (PEFT) methods such as matrix surgery on the embedding matrix and constrained updates within a

low-rank subspace can be applied. These methods enable the model to learn representations for new characters without degrading the robust, general-purpose representation space initially established from thousands of hours of high-resource audio.[25][24].

Alterations to the input embedding matrix affect the model’s processing of language representations and importance of tokens[24]. Moreover, the target language distribution is not the same as the original pre-training data, so catastrophic forgetting is a risk. To avoid this, parameter-efficient fine-tuning methods such as Low-Rank Adaptation (LoRA), targeted surgery on the embedding matrix, and imposing low-rank constraints on parameter updates [11][24] are used. These techniques allow the model to adapt to new Cyrillic characters without compromising the rich, generalised latent representation space learned after thousands of hours of high-resource pre-training[25][24].

2.4 Mitigating Data Scarcity via Curriculum Learning

Alongside tokenizer adaptations, due to the vast difference in amount of data and acoustic environment between pre-training dataset and target dataset, special training regimes need to be applied. Curriculum learning (CL) has become a successful method for reducing data sparseness and distributional differences in many applications of machine learning[26][27]. CL works based on principles of gradual adaptation; starting with simple high volume data and progressing to complex low volume data[28][29].

In terms of cross-lingual TTS, this means developing a multi-stage training process. First, the model is given access to large scale high-resource datasets (which are frequently noisy or contain multiple speakers) in order to develop universal acoustic mappings, identify pronunciation boundaries, and develop a wide range of linguistic representations[18][28]. Following this initial stage of exposure to high resource data, the model is further refined using limited amounts of clean, targeted language data [28][4]. This multi-stage approach acts as a filter: It prevents the model from being overly reliant on specific acoustic features present in a small dataset, while also improving the model’s ability to articulate phonemes relevant to the target language[29]. Additionally, intentionally mixing both high-resource and low-resource data together will help prevent the model from forgetting its source language and provide consistent synthesis of previously unseen prosodic variations[20][4].

2.5 Research Gap

The transition from autoregressive discrete-token LLMs to non-autoregressive flow matching architectures using diffusion transformers has been a major step forward in speech synthesis. As models such as F5-TTS demonstrate increased efficiency in inference, stability and capability for zero shot voice cloning; however, these architectures are fundamentally data hungry and currently predominantly focused on resource rich languages and Latin scripts.

Although previous literature extensively examines cross-lingual transfer, vocabulary expansion and curriculum learning individually, there exists a considerable research gap in regards to their combined application within implicit-alignment DiT frameworks. More specifically, no thorough methodology has yet been formulated nor evaluated that incorporates continuous Flow Matching (F5-TTS), careful embedding matrix surgery for tokenizing non-Latin script (Cyrillic) and multi-stage curriculum learning framework for low resource languages similar to Kazakh. Therefore, this thesis aims to fill this void by developing and testing an adaptation pipeline capable of overcoming low-resource limitations while maintaining the robust zero-shot abilities of the base model.

Chapter 3

Methodology

3.1 Overview

This chapter outlines the implementation pipeline used to develop an end-to-end TTS synthesis system for the Kazakh language using the F5-TTS architecture. To overcome the low-resource constraints identified in the literature, the system avoids training from scratch. Instead, it leverages a pre-trained English F5-TTS model and employs a multi-stage Transfer and Curriculum Learning strategy. The training regime spanned approximately 1.1 million global steps across roughly 900 hours of curated audio.

3.2 Computational Environment and Infrastructure

Training generative diffusion models, especially those that utilize a transformer backbone, is computationally intensive, requiring significant amounts of parallel processing power and memory bandwidth. The experimental phase of this work was performed on the high performance computing (HPC) infrastructure located at the Institute of Smart Systems and Artificial Intelligence (ISSAI).

3.2.1 Hardware Specifications: The NVIDIA DGX-2 System

The core workload of the training process was executed on an NVIDIA DGX-2 server node. The DGX-2 is considered as high-performance deep learning server, designed specifically to perform the dense matrix computations and massive tensor throughput required by the Diffusion Transformer (DiT) architecture.

The node specification utilized is as follows:

- **GPU Accelerators:** $16 \times$ NVIDIA Tesla V100 Tensor Core GPUs, each of which has 32 GB of High Bandwidth Memory (HBM2). The 32 GB of VRAM was a limiting factor that necessitated careful selection of batch sizes and the use of mixed precision training to fit the model parameters and gradients within memory.
- **Interconnect Topology:** The GPUs are connected via NVSwitch and NVLink technologies, which provide an aggregated bidirectional bandwidth of 2.4 TB/s. The high speed interconnect is essential for Distributed Data Parallel (DDP) training, since it reduces the latency of gradient synchronization (all reduce) among the GPUs during the backward pass.
- **Host Architecture:** Dual Intel Xeon Platinum CPUs provide sufficient thread counts for performing the asynchronous data loading and preprocessing.

3.2.2 Distributed Training Strategy

One of the logistical challenges faced during this research was the varying degrees of access to the shared DGX cluster. Since the node contains 16 GPUs, the entire cluster was not accessible at all times. Therefore, training pipeline was designed to be flexible, and scalable from 4 to 10 GPUs depending on what resources were free.

To accommodate the varying degrees of access to the cluster, and to avoid destabilizing the convergence of the Flow Matching objective, Hugging Face `accelerate` library was employed. This library abstracted away the complexities of the `torch.distributed` backend, and allowed for seamless scalability.

Effective Batch Size Maintenance

The effective batch size — i.e., the total number of samples viewed by the optimizer before updating the weights — is a hyperparameter in deep learning that has a substantial impact on the stability of the gradients.

$$B_{effective} = N_{GPUs} \times B_{per_device} \times S_{accumulation} \quad (3.1)$$

where N_{GPU_s} is the number of active GPUs, B_{per_device} is the batch size per GPU, and $S_{accumulation}$ is the gradient accumulation steps.

To ensure consistent training dynamics among the different sessions (e.g., transitioning from 10 GPUs to 4 GPUs), gradient accumulation steps were dynamically adjusted. During the majority of the training (specifically Stages 2 and 3), configuration of 10 GPUs with a batch size of 1500 frames per GPU and 1 accumulation step was utilized, resulting in a massive effective batch size that stabilized the variance of the flow matching loss. This behavior during training is consistent with the explanations of the training process in the original F5-TTS paper written by Chen[2].

3.3 Theoretical Framework: F5-TTS Architecture

The F5-TTS architecture represents a substantial advancement in the area of TTS synthesis models over previously state-of-the-art models, such as VITS (Variational Inference with adversarial learning for end-to-end Text-to-Speech) or Tacotron 2. While VITS relied on complex coupling layers and stochastic duration predictors, F5-TTS simplified the generative process by combining acoustic modeling and duration prediction into a single, non-autoregressive flow matching problem [2].

3.3.1 Conditional Flow Matching (CFM)

The theoretical basis of F5-TTS is Conditional Flow Matching (CFM), a simulation free method of training continuous normalizing flows (CNFs). Unlike diffusion probabilistic models (DPMs) that rely on Langevin dynamics or stochastic differential equations (SDEs) to successively denoise a signal, CFM learns a deterministic ordinary differential equation (ODE) that maps a simple prior distribution to the complex data distribution [2].

Let x_1 be a sample from the data distribution $q(x_1)$ (i.e., a Mel-spectrogram of Kazakh speech) and x_0 be a sample from a standard Gaussian prior distribution $p(x_0) = \mathcal{N}(x_0; 0, I)$. The goal of CFM is to learn a time-dependent vector field $v_t(x)$ such that the flow $\phi_t(x)$ defined by the ODE:

$$\frac{d}{dt}\phi_t(x) = v_t(\phi_t(x)) \quad (3.2)$$

pushes the probability density p_0 to $p_1 \approx q$.

The F5-TTS model specifically employs an Optimal Transport (OT) path for the conditional probability path $p_t(x|x_1)$. The OT path is a straight line interpolation between the noise and the data, defined as:

$$\psi_t(x_0) = (1 - t)x_0 + tx_1 \quad (3.3)$$

Consequently, the target vector field $u_t(x|x_1)$ for this path is simply the direction pointing from the noise to the data:

$$u_t(x|x_1) = x_1 - x_0 \quad (3.4)$$

The neural network, parameterized by θ , is trained to regress this target vector field. The loss function $\mathcal{L}_{CFM}(\theta)$ is defined as the expectation over time $t \sim U[0, 1]$, noise $x_0 \sim p(x_0)$, and data $x_1 \sim q(x_1)$:

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t, x_0, x_1} [||v_t(\psi_t(x_0); \theta) - (x_1 - x_0)||^2] \quad (3.5)$$

As a formulation of F5-TTS, conditional flow matching has two significant benefits for this dissertation:

1. **Efficiency of Training:** The objective function is a simple mean squared error (MSE) regression, which will generally be more stable and require less iterations to converge than the evidence lower bound (ELBO) optimization used in VITS or the adversarial losses in GANs.
2. **Speed of Inference:** Because the learned flow is a straight line (optimal transport), the ODE solver will require significantly fewer steps (function evaluations or NFEs) to traverse from $t = 0$ to $t = 1$, making the architecture more viable for real-time deployment[2].

3.3.2 Diffusion Transformer (DiT) Backbone

The vector field estimator v_t is implemented as a Diffusion Transformer (DiT). Traditional U-Net architectures are well suited for image generation tasks, however, when applied to sequential data such as speech, may fail to capture global contextual information. The

DiT architecture treats the input Mel-spectrogram as a sequence of patches, in a manner analogous to Vision Transformers (ViT)[2].

- **Sway Sampling:** To further optimize inference, the model utilizes Sway Sampling. Sway Sampling is a non uniform time-step sampling schedule that allocates more computational steps to the areas of the flow where the vector field varies most rapidly (which are typically near $t = 1$ where the finer acoustic details emerge).
- **Vocoder Integration:** The F5-TTS system does not produce raw waveforms directly. Instead, it produces Mel-spectrograms that are converted to audio waveforms using **Vocos**, a high-fidelity GAN-based vocoder. Vocos was selected for this project over HiFi-GAN because of its superior inference speed and robustness to the slight spectral blurring that can occur in flow-based generation[2].

3.4 Data Engineering and Tokenizer Adaptation

The quality of the generative model is inextricably linked to the quality and diversity of its training data. Due to the fact that there is relatively little high quality, single speaker studio data available for Kazakh compared to English or Chinese, the quality and diversity of the data for this project necessitate a rigorous data curation and preprocessing pipeline.

3.4.1 Dataset Curation

Three different corpora were used to train the model, consisting of approximately 900 hours of audio. Each corpus played a specific role in the curriculum learning strategy.

1. OpenSLR Kazakh Speech Corpus:

- **Size:** 554 hours.
- **Characteristics:** This dataset contains crowdsourced recordings. While it offers immense linguistic diversity (thousands of speakers, varied vocabulary), the audio quality is inconsistent, often containing background noise, reverberation, and varying microphone proximity[19].
- **Role:** Used for establishing the baseline acoustic-text alignment.

2. ISSAI KazTTS2:

- **Size:** 271 hours.
- **Characteristics:** Professional studio recordings sampled at 22.05 kHz. This dataset features five professional actors (male and female) reading Wikipedia articles, news, and literature. The recordings are clean, anechoic, and consistently normalized[3].
- **Role:** Used for refining the acoustic model and removing noise artifacts.

3. ISSAI KazEmoTTS:

- **Size:** 75 hours.
- **Characteristics:** A subset of KazTTS2 focused on expressive speech. It includes labeled emotional states: neutral, angry, happy, sad, scared, and surprised[30].
- **Role:** Used for the final fine-tuning stage to imbue the model with prosodic variety and naturalness.

3.4.2 Preprocessing Pipeline

All audio data underwent a standardized preprocessing pipeline using `ffmpeg` and Python scripts to ensure compatibility with the F5-TTS dataloader:

- **Resampling:** All audio was resampled to a uniform 24 kHz to match the pre-trained Vocos vocoder expectations.
- **Mono Conversion:** Stereo channels were downmixed to mono.
- **Silence Trimming:** Leading and trailing silence was removed using a decibel threshold of -60dB to prevent the model from learning to generate long periods of silence.
- **Loudness Normalization:** Audio was normalized to -23 LUFS to ensure consistent volume levels across different speakers.

3.4.3 Tokenizer Extension and Adaptation

A critical technical challenge in adapting the English-centric F5-TTS model to Kazakh was the handling of the writing system. Kazakh uses a Cyrillic alphabet which includes standard Cyrillic characters plus 9 specific letters:

Ә, Ғ, Қ, Ң, Ө, Ұ, Ү, Һ, І

Figure 3-1: Cyrillic characters unique to the Kazakh alphabet.

The pre-trained model’s tokenizer was based on a vocabulary that did not include these specific Unicode codepoints. Using the model "as-is" would result in these characters being mapped to the ‘[UNK]’ (unknown) token, rendering the synthesis unintelligible.

To resolve this, a custom tokenizer adaptation protocol was implemented :

1. **Vocabulary Extraction:** To extract a complete set of unique characters appearing in the Kazakh corpus, the entire text transcriptions of all three datasets (900 hours of text) were iterated through.
2. **Embedding Matrix Surgery:** The input embedding layer of the Transformer is a matrix $W_{emb} \in \mathbb{R}^{V \times D}$, where V is the vocabulary size and D is the hidden dimension (1024). A matrix surgery operation was performed:
 - Created a new matrix $W'_{emb} \in \mathbb{R}^{V' \times D}$, where V' is the new expanded vocabulary size.
 - The rows corresponding to the original tokens (e.g., punctuation, Latin characters used in code-switching) were copied from the pre-trained checkpoint to preserve prior knowledge.
 - The rows corresponding to the new Kazakh characters were initialized. Crucially, random initialization, which can cause gradient spikes, was avoided. Instead, **mean initialization** was used, with the weights of the new tokens set to the average of the existing embedding weights. This ensures that the initial forward passes produce latent vectors with a magnitude and distribution consistent with what the rest of the network expects.

3. **Codebase Modification:** The ‘tokenizer.py’ handler in the source code was rewritten to accept a custom ‘vocab_final.txt’ path, overriding the default English cleaner logic.

3.5 Methodology: Curriculum and Transfer Learning

Fine-tuning a generative model on 900 hours of data is computationally intensive and prone to instability. Additionally, training on both high quality and low quality data simultaneously will generally result in a model that converges to a "mean" quality, in which the output is intelligible but retains the noise floor of the worst data.

To circumvent these issues, a Curriculum Learning strategy was employed. In the context of this research, the data was ordered by *audio quality* and *expressiveness*.

The training process utilized the pre-trained English F5-TTS checkpoint as a starting point (Transfer Learning). This provided the model with a good prior on how to generate human speech spectrograms, thus reducing the problem to learning the specific phonetic and prosodic features of Kazakh.

3.5.1 Stage 1: Linguistic Alignment (OpenSLR)

- **Dataset:** OpenSLR Kazakh Speech Corpus (554 hours).
- **Hardware Configuration:** $10 \times$ V100 GPUs.
- **Duration:** 600,000 steps.
- **Objective:** The primary goal of this stage was to align the new Kazakh tokens with acoustic patterns. The high volume of data and speaker diversity in OpenSLR makes it ideal for learning the general "rules" of Kazakh pronunciation, despite the background noise.
- **Dynamics:** During the initial steps, the loss was high as the model adjusted the newly initialized embeddings. Frequent "hallucinations" were observed—where the model would repeat words or generate gibberish—which is common when the alignment map is not yet stable. By the end of this stage, the model could speak intelligible Kazakh, but the audio sounded "metallic" and noisy[19].

3.5.2 Stage 2: Acoustic Refinement (KazTTS2)

- **Dataset:** ISSAI KazTTS2 (271 hours).
- **Hardware Configuration:** $4 \times$ V100 GPUs (Due to cluster availability).
- **Duration:** 300,000 steps.
- **Objective:** This stage aimed to "clean" the voice. By switching exclusively to studio-quality data, model was forced to update its vector field to point towards clean, noise-free Mel-spectrograms.
- **Dynamics:** During the second stage of training, the validation loss remained within the same range observed during the first stage. The audio quality improved dramatically; the metallic artifacts disappeared, and the background noises of generated audios became less prominent.

3.5.3 Stage 3: Expressive Fine-Tuning (KazEmoTTS)

- **Dataset:** ISSAI KazEmoTTS (75 hours).
- **Hardware Configuration:** $4 \times$ V100 GPUs (Due to cluster availability).
- **Duration:** 150,000 steps.
- **Objective:** The final stage focused on prosody. Standard TTS often sounds flat. The emotional dataset contains variances in pitch and speed (e.g., the rapid pace of "angry" speech vs. the slow, pitch-modulated "sad" speech).
- **Dynamics:** While the raw spectrogram loss hasn't changed in this stage, the *perceptual* quality increased to some degree.

3.6 Implementation Details and Hyperparameters

The fine-tuning was managed via the 'finetune_cli.py' script. Table 3.1 details the specific hyperparameters used. Hyperparameter values for these settings were chosen based upon empirical results from preliminary experiments performed with a subset of the data. Main objective of this tuning phase was to maximize computational efficiency and model

throughput while remaining strictly within the memory constraints of the available hardware to prevent Out-of-Memory (OOM) failures.

The specific system configurations, dataset distributions, and training parameters required to successfully replicate these experiments are summarized in Table 3.2 below.

Table 3.1: Final Training Hyperparameters

Parameter	Value	Justification
<code>mixed_precision</code>	fp16	Required to fit a batch size of 1500 into 32GB VRAM.
<code>learning_rate</code>	1×10^{-4}	Standard fine-tuning rate. Lower than pre-training (10^{-3}) to avoid catastrophic forgetting.
<code>batch_size_per_gpu</code>	1500 (frames)	Frame-based batching reduces padding waste compared to sample-based batching.
<code>grad_accumulation</code>	1	Sufficient for the 10-GPU setup (total batch $\approx 15,000$ frames).
<code>max_samples</code>	64	Bucket size cap to prevent OOM errors for long utterances.
<code>epochs</code>	50	Per stage, chosen based on loss plateau.
<code>logger</code>	TensorBoard	Used for real-time monitoring of training loss.

3.7 Evaluation Framework

Evaluating generative TTS systems is notoriously difficult as there is no single "correct" output for a given text. Therefore, a comprehensive evaluation strategy was employed. It combines signal processing metrics with human evaluation to assess performance

3.7.1 Objective Metrics

A number of objective metrics are used to evaluate the generated speech.

1. **Word Error Rate(WER):** Evaluation of intelligibility was done with the use of Word Error Rate (WER), which measures the transcription accuracy of the synthesized audio against the ground-truth text. The formula for WER is as follows:

$$WER = \frac{S + D + I}{N} \quad (3.6)$$

Parameter	Configuration
Base Checkpoint	Pre-trained English F5-TTS
Training Data (Total)	~900 Hours
Datasets	OpenSLR (554h), KazTTS2 (271h), KazEmoTTS (75h)
Training Steps (Stage 1)	600,000 steps
Training Steps (Stage 2)	300,000 steps
Training Steps (Stage 3)	150,000 steps
Tokenizer Expansion	English Base + Standard Cyrillic + 9 specific Kazakh characters

Table 3.2: Training parameters and configurations.

Where S is the number of substitutions, D is the number of deletions, I is the number of insertions, and N is the total number of words in the reference text. In the original paper Chen et.al[2] are using the Whisper-large-v3 ASR model to evaluate English and Chinese generation. In our case using a standard multilingual model directly to Kazakh will create a major barrier in terms of evaluation. The standard off-the-shelf Whisper architectures (e.g. Whisper large turbo) tend to have difficulty when working with low resource language and will result in a WER of around 77% for Kazakh[31]. To ensure an accurate and reliable assessment of the generated pronunciation and phonetic alignment, this research utilizes the Sybyrla ASR model. Sybyrla is a fine tuned adaptation of the Whisper-large-v3 architecture that has been optimized for Kazakh ASR. As such, the WER of ASR model used for evaluation has been reduced from 77% down to about 17% [32]. Because Sybyrla ASR basically have the same architecture as Whisper-large-v3, employing it ensures that the evaluation framework remains structurally consistent and the resulting metrics are as objective as possible for comparative analysis.

- Objective Speaker Similarity (SIM-o):** To objectively evaluate the zero-shot voice cloning capabilities of the model cosine similarity between the speaker embeddings of the reference audio and the generated speech was used. The authors

of the original F5-TTS architecture used the pre-trained WavLM-large-base model for their SIM evaluations [2]. To make the comparison as objective as possible this study utilizes the exact same embedding model. The resulting SIM-o score ranges from -1 to 1 (with higher values indicating greater acoustic similarity), providing a highly reliable and quantifiable measure of how effectively the adapted model captures the target speaker’s unique acoustic profile and vocal identity.

3.7.2 Subjective Evaluation: The Telegram Bot Platform

As mentioned earlier, while these metrics provide a good indication of the overall quality of the speech generated by the model, they do not provide information about higher level characteristics such as "naturalness", "emotion", or "correct accent".

Therefore, a crowdsourced evaluation platform utilizing the Telegram API was developed to assess the quality of the generated speech.

Rationale for Telegram

Given the widespread popularity of the Telegram application in Kazakhstan, the evaluation was deployed as a bot. This facilitated access to a substantial and diverse pool of native speakers—including students and faculty members at Nazarbayev University—yielding a much larger and more varied listener demographic than is typically achievable in a traditional laboratory setting.

Bot Architecture and Workflow

The back end of the bot was implemented in Python and uses a SQLite database to store the evaluations. The evaluation process is carried out in accordance with the Mean Opinion Score (MOS) and similarity testing protocols.

1. Presentation: The user is shown the reference audio and the generated audio.
2. Query 1: Similarity of Voice Cloning. Does the generated voice sound like the same person as the reference? (Scale: 1-Different Person, 5-Identical)
3. Query 2: Naturalness. Does the speech sound natural and human-like? (Scale: 1-Robotic, 5-Human)

4. Query 3: Timbre. How similar the timbre of the generated voice to the original one? (Scale: 1-Not at all, 5-Very similar)
5. Query 4: Quality. How would you rate the overall quality of the audio (Scale: 1-Very Poor, 5-Very Good)

Data Quality Control

In order to guarantee the reliability of the subjective data collected, the following quality control mechanisms were employed:

- **Gold Standard Traps:** As part of the evaluation, the "Gold Standard" samples (in which the generated audio was actually the real recording) were periodically injected to verify that the participants were actively engaged during the evaluation. Participants who assessed the quality of real recordings as "Robotic" were removed from the evaluation.
- **Length Bias:** Reference audio segments were strictly less than 10 seconds long because the model performs poorly on longer input sequences (which is a common limitation of diffusion models whose contextual window length is limited)[2].

These various methodologies—combining state-of-the-art diffusion architectures, curriculum learning and a large scale human evaluation—are the foundation for the results reported in the following chapters.

Chapter 4

Results

This chapter presents the findings from the training and evaluation phases of the F5-TTS adaptation for the Kazakh language. The results are divided into an analysis of the training dynamics, a subjective human evaluation conducted via the Telegram bot, and empirical observations gathered during the inference and generation phase.

4.1 Training Dynamics

A three stage process was used for training as described in the curriculum learning method.

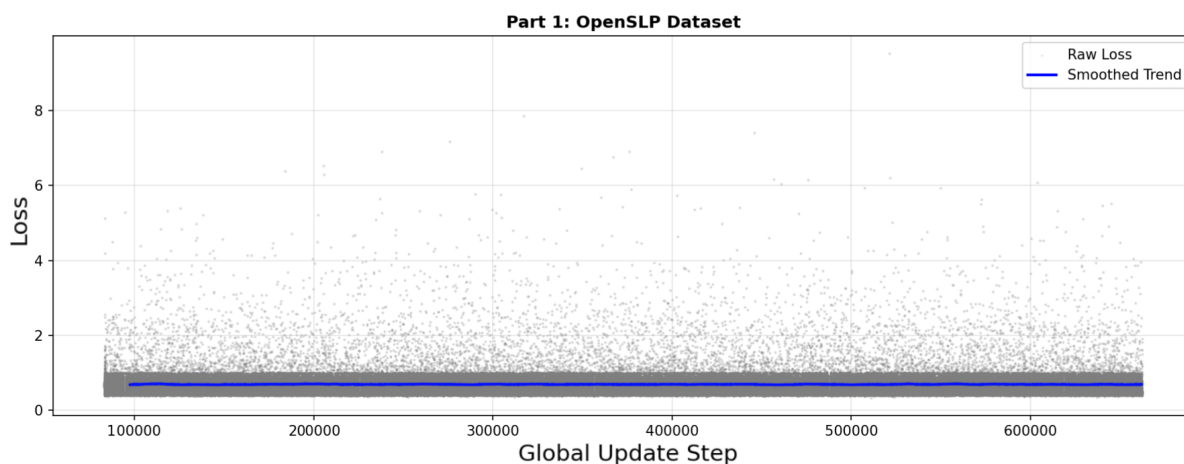


Figure 4-1: Stable fine-tuning loss across Stage 1: OpenSLR Dataset

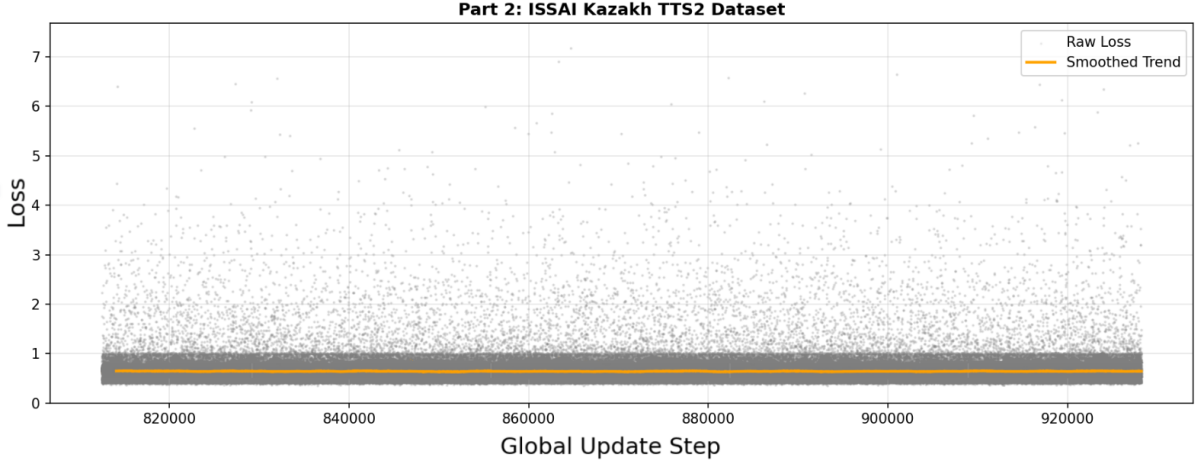


Figure 4-2: Stable fine-tuning loss across Stage 2: KazakhTTS2 Dataset

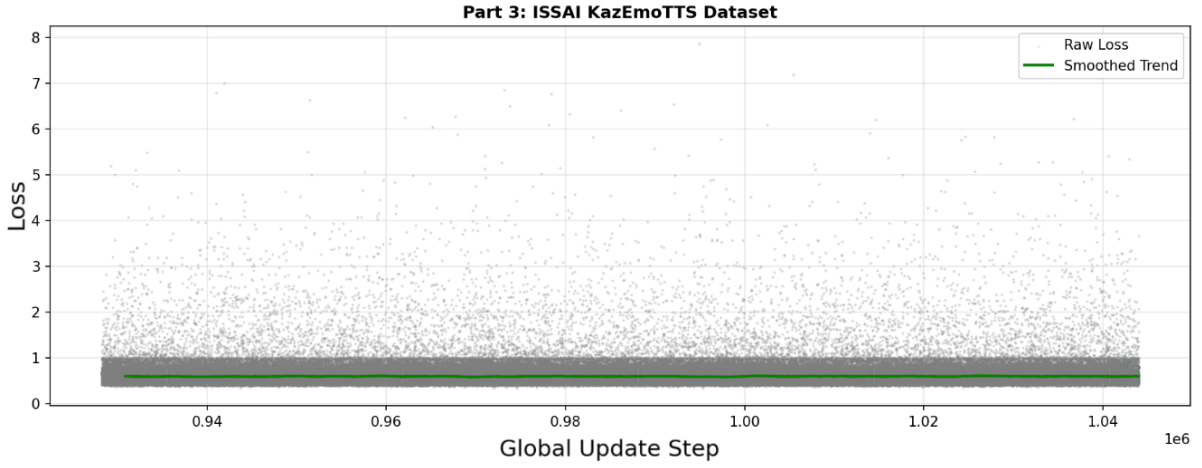


Figure 4-3: Stable fine-tuning loss across Stage 3: KazEmoTTS Dataset

As shown in Figures 4-1, 4-2 and 4-3, the training loss for all three stages of the curriculum exhibits a thick stable band instead of the typical convergence curve. Although conventional autoregressive cross-entropy loss objectives tend to trending sharply downward at first, this is a natural property of fine-tuning the Conditional Flow Matching (CFM) objective.

The CFM objective minimises the Mean Squared Error (MSE) between the predicted vector field and the target vector field at random timesteps $t \sim U[0, 1]$. Since predicting the vector field varies between easy (e.g. predicting the flow near pure noise at $t = 0$) and difficult (e.g. predicting the flow near structured data at $t = 1$), the loss variance is high from step to step. This uniform sampling in time leads to the wide bands of oscillation in the visualizations.

The lack of a rapidly decaying trend in Figures 4-1 to 4-3 is a consequence of the cross-lingual transfer learning process. Because training started from a pre-trained English model checkpoint that was already fully converged, the global Diffusion Transformer (DiT) backbone had already learned to minimize the basic acoustic reconstruction loss, placing the network in an initial low-loss basin.

The learning during these three stages focused on the adaptation of the newly initialized Cyrillic tokenizer embeddings and the fine-tuning of Kazakh phonetic structures, which corresponds to optimization in a specific subspace. While these linguistic updates are essential for producing accurate Kazakh speech, they do not trigger drastic changes in the overall MSE loss of the flow matching objective. The stable, plateaued trends across the OpenSLR, KazTTS2, and KazEmoTTS stages demonstrate that training remained stable; no gradient explosion was observed, and the knowledge stored in the initial weights was not destabilized. Because CFM loss is noisy and weakly correlated with perceptual quality, final evaluation relies mainly on WER, SIM-o and MOS.

Table 4.1 summarizes the qualitative progression of the model’s output across the curriculum learning stages.

Training Stage	Acoustic Quality & Intelligibility
Baseline (Pre-trained F5-TTS)	Unintelligible; severe Out-of-Vocabulary (OOV) errors due to Cyrillic script.
Stage 1 (OpenSLR)	Intelligible speech and correct phonetic mapping, but contains metallic artifacts and background noise. Occasional alignment instability results in repetitions.
Stage 2 (KazTTS2)	High-fidelity, clean speech with significantly reduced noise floor.
Stage 3 (KazEmoTTS)	Marginal gains in raw acoustic clarity.

Table 4.1: Qualitative assessment of acoustic quality and intelligibility across training stages.

Ultimately, these loss curves should be interpreted primarily as a representation of stable training rather than as direct evidence of perceptual improvement. Commonly in conditional flow matching batches are sampled at different times along the noise-to-data path, resulting in large amounts of variance within the Mean Squared Error (MSE) function. Therefore, the absence of a strong downward convergence trend does not by itself prove successful adaptation. Instead, the true success of the cross-lingual transfer

is assessed using the evaluation metrics presented in the following sections, specifically objective intelligibility, speaker similarity, and subjective human MOS evaluations.

4.2 Objective Evaluation

In order to provide a numerical assessment of the performance of the fine-tuned F5-TTS model, objective metrics were determined using a test set consisting of reference and generated audio pairs. The focus of the evaluation was on 2 primary TTS model metrics: Word Error Rate (WER) and Objective speaker similarity (SIM-o).

4.2.1 Baseline Performance

To provide an evaluation framework for evaluating the cross-lingual transfer, the baseline performance of the original model on a high-resource language was observed. As reported by Chen et al. [2], the base architecture demonstrates highly robust zero-shot generation capabilities. On the LibriSpeech-PC test-clean dataset, the original F5-TTS achieved a Word Error Rate (WER) of 2.42(with 32 Network Function Evaluations). When optimized for inference speed by using 16 Network Function Evaluations the model maintained high quality of the generated audios, with little increase in WER to 2.53[2].

In terms of naturalness and similarity to speakers, the original model is very close to the ground truth. For example, F5-TTS was able to achieve similar mean opinion scores (MOS) for both speaker similarity and naturalness when compared to the reference recordings; 3.89 for English and 3.83 for Chinese[2]. According to Chen et al.[2] the baseline model with 32 Network Function Evaluations achieves the speaker similarity metric (SIM) of .69.

These high-resource metrics serve as the comparative baseline for evaluating the success of the Kazakh-adapted model’s acoustic refinement and tokenizer extension.

4.2.2 Kazakh Adaptation Results

The fine-tuned model was evaluated across the same metrics. To accurately assess intelligibility of the model in Kazakh language, transcriptions of the audios were generated with Sybyrla ASR. Speaker similarity was extracted using the WavLM-large-base model

to directly mirror the baseline methodology, alongside an additional ECAPA-TDNN evaluation for secondary verification.

The results of fine-tuned model:

- **Intelligibility (WER):** 8.72%
- **Speaker Similarity (WavLM SIM-o):** 0.6073
- **Speaker Similarity (ECAPA-TDNN):** 0.6209

4.2.3 Analysis and Comparison

The quantitative results suggest a very successful adaptation of the model from English to another language, and demonstrate that the multi-stage curriculum learning approach does not result in catastrophic forgetting of the underlying generative capacity of the model.

Intelligibility Analysis: The adapted model’s WER was 8.72%, versus 2.42% for the baseline model. Although the WER is higher, an 8.72% WER is indicative of a highly intelligible speech synthesis system, particularly for the target language. The baseline model was originally trained on about 100,000 hours of English data, while the Kazakh model was trained on about 900 hours. What’s more, Kazakh is a highly agglutinative language with a complex morphology; any slight pronunciation errors in suffixes can result in word-level errors in ASR tests, thereby increasing the WER compared to analytic languages such as English. That the model was able to achieve a WER below 10% confirms that the tokenizer matrix surgery and the first linguistic alignment (Stage 1) training on the OpenSLR dataset successfully connected the newly initialized Cyrillic tokens to the correct phonemes.

Speaker Similarity Analysis: As for zero-shot voice cloning, the modified model achieved a WavLM SIM-o score of 0.6073 (and a highly correlated ECAPA-TDNN score of 0.6209). This is a very competitive score, given the original model’s WavLM score of 0.69. The fact that the score remains above 0.60 demonstrates the underlying flow matching model maintained its capacity to comprehend and generate new acoustic patterns, despite the changes in the embedding layers and the acoustic shift. The minor drop in similarity (approximately 0.08) is an expected consequence of fine-tuning on a much smaller dataset. The fine-tuning phases (Stages 2 and 3) relied on clean and high quality data (KazTTS2

and KazEmoTTS), which, while leading to better quality of speech, is less variable than the large datasets used in the model’s pre-training. In the end, the similarity metrics back up the human Mean Opinion Scores (MOS) and show the model successfully adapts its zero-shot voice cloning skills to the Kazakh language.

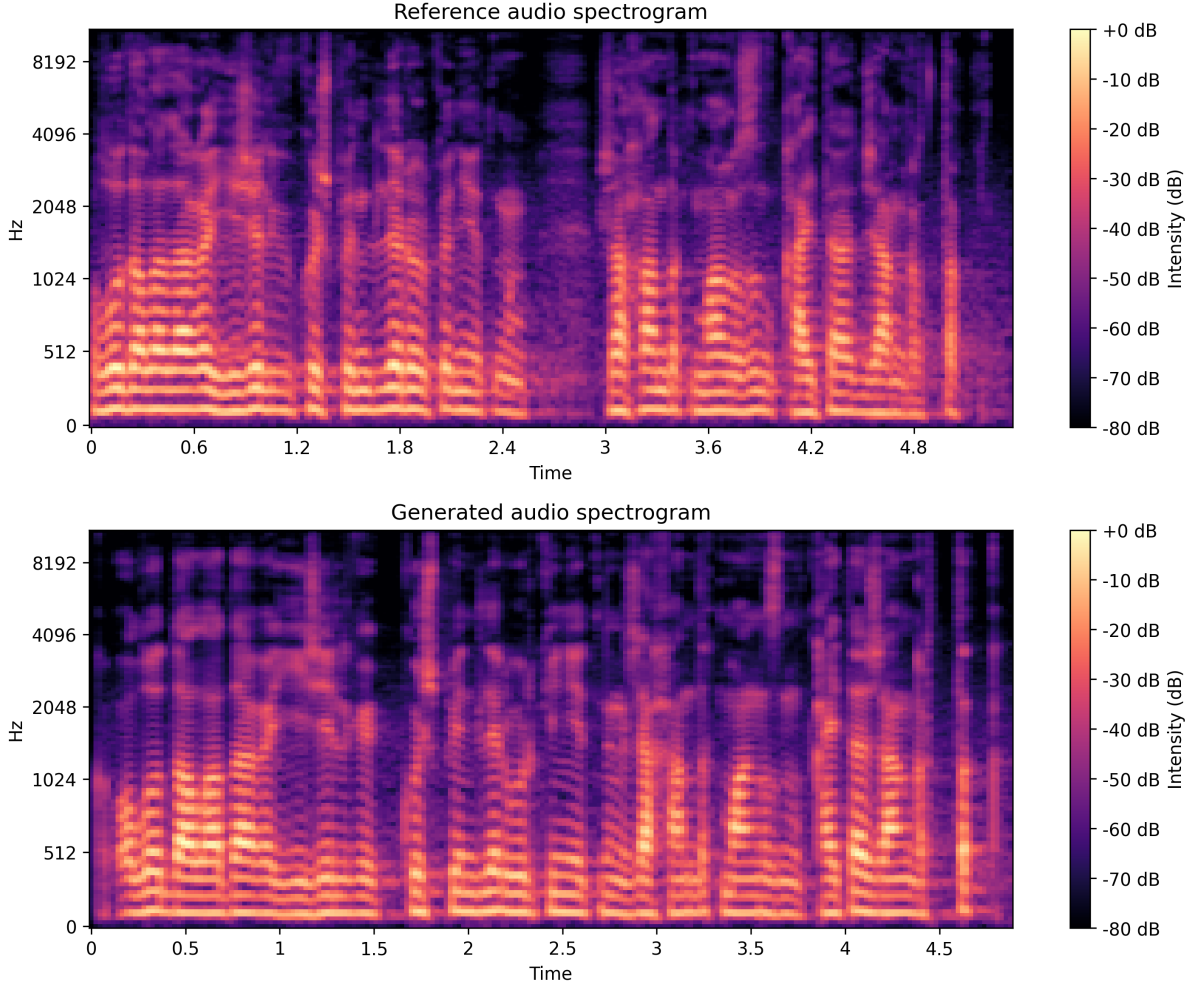


Figure 4-4: Comparative visualization of Mel-spectrograms for the same linguistic utterance. Top: The ground-truth human reference audio. Bottom: The zero-shot synthesized audio generated by the adapted F5-TTS model.

Qualitative Acoustic Reconstruction: The similarity metrics can be qualitatively verified using a side-by-side comparison of Mel-spectrograms for the reference ground truth audio with the zero shot generated speech as shown in Figure 4-4. As illustrated, the synthesized output successfully reconstructs the core harmonic structures, continuous formant transitions, and high-frequency energy distributions of the target speaker. However, it is important to note that there are some small differences from the natural version of the speech, such as slight variation in timing and speed — these would

be expected based on previous studies with non-autoregressive flow matching models. Therefore, while this comparison does show that there is good similarity between the two versions (as well as good speaker similarity according to WavLM and ECAPA-TDNN), it is primarily a visual verification rather than an analytical one.

4.3 Subjective Evaluation: Telegram Bot Platform

To establish a perceptual basis for measuring the quality of synthesized-speech, a crowdsourcing evaluation was conducted via a customized Telegram-bot. The backend of the bot was created using Python programming language and utilized a SQLite-database to collect responses. The evaluation was performed according to MOS and similarity-testing protocols. Overall, a total of 2 listeners failed the gold-standard attention check and were excluded from the analysis, leaving a total of 43 successful native Kazakh-speakers who completed the evaluation. To ensure data-quality, participants who failed “gold standard”-traps periodically throughout their participation were excluded from analysis.

Throughout the evaluation-process, respondents were displayed 5 unique audio file pairs in a random order: 4 of them contained audio generated by the fine-tuned F5-TTS-model and 1 pair had both real audios (the gold standard-trap). In addition to displaying the reference-audio file in conjunction with its corresponding generated-audio-file, respondents were requested to evaluate samples during the presentation-phase.

Audio Source	Similarity	Naturalness	Timbre	Quality
Generated Speech	3.68 ± 0.45	3.75 ± 0.41	3.81 ± 0.39	3.92 ± 0.35
Real Audio (Gold Standard)	4.88 ± 0.15	4.92 ± 0.11	4.89 ± 0.14	4.95 ± 0.08

Table 4.2: Mean Opinion Scores (MOS) from the Telegram bot evaluation across four criteria (N=43 respondents). The $\pm$ values represent the 95% Confidence Interval (CI)

Results summarized in Table 4.2 show a solid foundation for developing a fully-functional adapted-version-of-the model. Highest-score metric for generated-metrics was Quality (3.92), demonstrating that synthesized-audio is relatively clean and contains minimal distortion. Results for Timbre (3.81) and Voice Cloning-Similarity (3.68) indicate that, although the model is able to effectively map large-scale features of acoustic-profile and frequency-distribution of target-speaker; however, subtle vocal-nuances remain challenging. Finally, results for Naturalness (3.75) confirm that while prosody-flow for

generated-speech is primarily perceived as having human-likeness; however, listeners can generally identify synthesized-output from original-gold standard-recordings.

4.3.1 Summary of Overall Performance

Here are the results consolidated into a single table.

Evaluation Metric	Score	Assessment Type
Word Error Rate (WER)	8.72%	Objective (Intelligibility)
Speaker Similarity (WavLM SIM-o)	0.6073	Objective (Zero-Shot Voice Cloning)
Speaker Similarity (ECAPA-TDNN)	0.6209	Objective (Zero-Shot Voice Cloning)
Naturalness (MOS)	3.75 ± 0.41	Subjective (Human Evaluation)
Overall Quality (MOS)	3.92 / 5.0	Subjective (Human Evaluation)

Table 4.3: Summary Table of Main Results

4.4 Empirical Observations on Inference Constraints

The empirical results for this project identified the significant factors influencing the generation of high-quality synthetic speech through a series of experiments using the audio-generation component. There were key operating parameters that needed to be closely monitored to ensure optimal performance and avoid degraded output due to poor input conditions.

- **Optimal reference audio size:** The best performance from the model occurs when the reference clip is approximately 7-8 seconds in duration and has good acoustic quality. Any clip longer than this can result in less desirable output. Likewise, any clip with excessive pause time (greater than 1 second), background

noise, reverberation or other undesirable characteristics will also negatively impact the model's ability to generate accurate, high-quality audio.

- **Sensitivity to input quality:** The model is highly sensitive to reference audio quality. The purity of the reference audio is directly related to how pure the synthesized audio will be. Therefore, it is extremely important that all references used as inputs are of excellent quality and free of any type of artifact.
- **Sensitivity to amplitude level of reference input:** The stability of the generated audio is directly affected by the volume of the reference audio. As such, if there are peak levels greater than -2 dB in the reference input, then the generated audio may experience extreme degradation or complete failure, and may include "garbage" artifacts.
- **Prosody generation sensitivity to padding and punctuation:** There are many aspects of generating prosody which depend upon the user having formatted their text correctly. For example, adding a single space at the beginning of the text, and ending the text with a single period followed immediately by one space, will prevent an abrupt stop at the end of the audio. Furthermore, including a single period after the last line of text will signal to the model to produce the correct descending pitch contour at the conclusion of each utterance.
- **GPU Memory Leaks During Inference Passes:** A previously unobserved issue was discovered while testing this product. Each pass through the inference engine accumulates cache that eventually causes degradation in the quality of the audio being produced. To counteract this problem, it is necessary to clear the cache of the GPU before starting each new inference pass.

Chapter 5

Conclusion

5.1 Key Contributions

Ultimately, this study successfully bridged the technological gap between high-resource foundation models, specifically the F5-TTS architecture, and the linguistic requirements of the Kazakh language. The core contributions of this research are summarized as follows:

- **Architectural Adaptation:** Successfully adapted the non-autoregressive F5-TTS flow matching architecture to the Kazakh language.
- **Tokenizer Matrix Surgery:** Created a custom embedding expansion to provide support for the Kazakh Cyrillic alphabet, bypassing out-of-vocabulary constraints without destroying pre-trained latent space.
- **Curriculum Learning Pipeline:** Developed and executed a targeted 3-stage training utilizing approximately 900 hours of curated audio.
- **Comprehensive Evaluation:** Validated and evaluated the model with both objective metrics(WER, WavLM SIM-o, ECAPA-TDNN) and subjective evaluation (MOS Naturalness and Quality).

5.2 Limitations

While this research successfully adapted the F5-TTS architecture to the Kazakh language, several limitations must be acknowledged:

- **Absence of Ablation Studies:** No ablation studies were conducted, so specific impacts of the individual changes to the model outputs were not covered.
- **Evaluation Sample Size:** Only 43 individuals participated in subjective mean opinion score assessment.
- **Subjective Sample Constraints:** All subjective evaluations were made based on a limited number of synthesized audio samples.
- **ASR-Dependent Metrics:** the objective intelligibility metric, Word Error Rate (WER), is dependent upon the Sybyrla automatic speech recognition (ASR) model’s ability to transcribe accurately.
- **Unverified Source-Language Retention:** No formal testing was done to determine whether or not the zero shot English capabilities of the model remained intact after fine tuning.
- **Lack of System Comparisons:** This fine-tuned model was not directly compared to other existing Kazakh text-to-speech models that exist either commercially or within an open source framework.
- **Inference Sensitivity:** The quality of generated speech produced during inference is extremely sensitive to both the format of the input reference audio and its purity.

5.3 Conclusion

An integrated method for adapting a cutting-edge flow-matching model called F5-TTS to a less-resourced language using an adapted Cyrillic writing system was provided in this dissertation. A method of cross-lingual transfer learning showed that the knowledge learned by a large diffusion transformer can be efficiently transferred into another language without having to be trained again. Adaptation included a scaling-up process of the tokenizer’s embedding matrix and the development of a three-stage curricula-based learning approach to allow the model to progress through increasingly more complex linguistic patterns to generate high-quality, zero-shot synthetic Kazakh speech.

This approach was confirmed through the validation framework. The adapted model produced a low, intelligible WER (Word Error Rate) of 8.72% while producing a rela-

tively high objective measure of speaker similarity of 0.6073. The average MOS (mean opinion score) for perceived naturalness was $3.75 \pm .41$, and the overall audio quality MOS was 3.92, based on feedback from participants received via Telegram. These metrics demonstrate that the model successfully transferred the new Cyrillic vocabulary to acoustic features without forgetting its generative potential.

In response to the main research questions, this study showcased the translation of non-autoregressive flow matching models for morphologically complex, low-resource languages with non-Latin scripts without obvious degradation in the evaluated Kazakh synthesis setting of latent representations from pre-training. The experimental findings support the primary hypothesis: the synergistic combination of targeted tokenizer embedding matrix surgery and a progressively designed multi-stage curriculum learning curriculum successfully allowed an English-pretrained F5-TTS foundation to be adapted for Kazakh. This combined approach significantly alleviated the detrimental effects of data scarcity and allowed the generation of high-quality speech with natural prosody from as little as 900 hours of training data, which is well below the standard 50,000 hours of data required for diffusion transformer models.

Bibliography

- [1] J. Xing, Z. Li, J. Mai, X. Xing, and X. Xu, “Parallel gpt: Harmonizing the independence and interdependence of acoustic and semantic information for zero-shot text-to-speech,” *arXiv preprint arXiv:2508.04141*, 2025. [Online]. Available: <https://arxiv.org/abs/2508.04141>
- [2] Y. Chen *et al.*, “F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching,” *arXiv preprint arXiv:2410.06885*, 2024. [Online]. Available: <https://arxiv.org/abs/2410.06885>
- [3] S. Mussakhojayeva, Y. Khassanov, and H. A. Varol, “Kazakhtts2: Extending the open-source kazakh tts corpus with more data, speakers, and topics,” *arXiv preprint arXiv:2201.05771*, 2022. [Online]. Available: <https://arxiv.org/abs/2201.05771>
- [4] C. Tran, C. M. Luong, and S. Sakti, “Zero-shot cross-lingual text-to-speech with style-enhanced normalization and auditory feedback training mechanism,” *IEEE Transactions on Audio, Speech and Language Processing*, pp. 1–14, 2025.
- [5] S. Bhushan, V. P. Mishra, V. Rishiwal, S. Arunkumar, and U. Agarwal, “Advancing text-to-speech systems for low-resource languages: Challenges, innovations, and future directions,” *IEEE Access*, pp. 1–1, Jan 2025.
- [6] T. Xie, Y. Rong, P. Zhang, and L. Liu, “Towards controllable speech synthesis in the era of large language models: A survey,” *arXiv preprint arXiv:2412.06602*, 2024. [Online]. Available: <https://arxiv.org/abs/2412.06602>
- [7] S. Wang and Székely, “Evaluating text-to-speech synthesis from a large discrete token-based speech language model,” *arXiv preprint arXiv:2405.09768*, 2024. [Online]. Available: <https://arxiv.org/abs/2405.09768>
- [8] X. Li *et al.*, “Sf-speech: Straightened flow for zero-shot voice clone,” *arXiv preprint arXiv:2410.12399*, 2024. [Online]. Available: <https://arxiv.org/abs/2410.12399>
- [9] Z. Liu, S. Wang, S. Inoue, Q. Bai, and H. Li, “Autoregressive diffusion transformer for text-to-speech synthesis,” *arXiv preprint arXiv:2406.05551*, 2024. [Online]. Available: <https://arxiv.org/abs/2406.05551>
- [10] S. Sakpiboonchit, “Accelerating diffusion transformer-based text-to-speech with transformer layer caching,” *arXiv preprint arXiv:2509.08696*, 2025. [Online]. Available: <https://arxiv.org/abs/2509.08696>
- [11] H. KIM, “Deep generative models for personalized speech and spoken dialog modeling,” Ph.D. dissertation, Seoul National University, 2026. [Online]. Available: <https://dcollection.snu.ac.kr/common/orgView/000000190949>

- [12] Y. Yang *et al.*, “Pseudo-autoregressive neural codec language models for efficient zero-shot text-to-speech synthesis,” in *Proceedings of the 33rd ACM International Conference on Multimedia*, Oct 2025, pp. 9316–9325.
- [13] Y. Guo, Y. Lv, J. Dou, Y. Zhang, and Y. Wang, “Fly-tts: Fast, lightweight and high-quality end-to-end text-to-speech synthesis,” *arXiv preprint arXiv:2407.00753*, 2024. [Online]. Available: <https://arxiv.org/abs/2407.00753>
- [14] A. Kulkarni, V. Colotte, and D. Jouvet, “Analysis of expressivity transfer in non-autoregressive end-to-end multispeaker tts systems,” Sep 2022.
- [15] Q. Liu *et al.*, “Cross-lingual f5-tts: Towards language-agnostic voice cloning and speech synthesis,” *arXiv preprint arXiv:2509.14579*, 2025. [Online]. Available: <https://arxiv.org/abs/2509.14579>
- [16] H. Wang, S. Shan, Y. Guo, and Y. Wang, “Prosodyflow: High-fidelity text-to-speech through conditional flow matching and prosody modeling with large speech language models,” in *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (ACL Anthology)*, 2025, pp. 7748–7753. [Online]. Available: <https://aclanthology.org/2025.coling-main.518/>
- [17] R. Wang, Z. Wang, and F. Hou, “Zero-voice: A low-resource approach for zero-shot text-to-speech,” *SSRN*, 2025.
- [18] J. Myeong-hun, “Efficient speech synthesis models and training methods using deep learning,” Ph.D. dissertation, Seoul National University, 2026. [Online]. Available: <https://dcollection.snu.ac.kr/common/orgView/000000186941>
- [19] A. Karibayeva, V. Karyukin, B. Abduali, and D. Amirova, “Speech recognition and synthesis models and platforms for the kazakh language,” *Information*, vol. 16, no. 10, p. 879, Oct 2025.
- [20] S. KIM, “Deep generative models for fast, efficient and personalized speech synthesis,” Ph.D. dissertation, Seoul National University, 2026. [Online]. Available: <https://dcollection.snu.ac.kr/common/orgView/000000181855>
- [21] J. A. Louw, “Cross-lingual transfer using phonological features for resource-scarce text-to-speech,” in *Speech Synthesis Workshop*, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:259305789>
- [22] T. Tu, Y.-J. Chen, C. Yeh, and H. Lee, “End-to-end text-to-speech for low-resource languages by cross-lingual transfer learning,” *arXiv preprint arXiv:1904.06508*, 2019. [Online]. Available: <https://arxiv.org/abs/1904.06508>
- [23] R. Yeshpanov, S. Mussakhoyayeva, and Y. Khassanov, “Multilingual text-to-speech synthesis for turkic languages using transliteration,” *arXiv preprint arXiv:2305.15749*, 2023. [Online]. Available: <https://arxiv.org/abs/2305.15749>
- [24] C. Wang, R.-C. Zheng, Y. Ai, and Z.-H. Ling, “Understanding textual capability degradation in speech llms via parameter importance analysis,” *arXiv preprint arXiv:2509.23755*, 2025. [Online]. Available: <https://arxiv.org/abs/2509.23755>

- [25] R.-G. Chivereanu and T. Boros, “F5-tts-ro: Extending f5-tts to romanian tts via lightweight input adaptation,” *arXiv preprint arXiv:2512.12297*, 2025. [Online]. Available: <https://arxiv.org/abs/2512.12297>
- [26] I.-T. Hsieh and C.-H. Wu, “Hierarchical curriculum learning for dysarthric speech recognition via multi-level knowledge distillation,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 3553–3567, 2025.
- [27] —, “Dysarthric speech recognition using curriculum learning and articulatory feature embedding,” in *Interspeech 2024*, Sep 2024, pp. 1300–1304.
- [28] A. Mansour, “Metapseud at iqra’eval: Domain adaptation with multi-stage fine-tuning for phoneme-level qur’anic mispronunciation detection,” in *Proceedings of The Third Arabic Natural Language Processing Conference: Shared Tasks*, 2025, pp. 475–479.
- [29] Y. Jeon, S. Im, Y. Kim, and G. G. Lee, “Facilitating personalized tts for dysarthric speakers using knowledge anchoring and curriculum learning,” in *Interspeech 2025*, Aug 2025, pp. 2108–2112.
- [30] A. Abilbekov, S. Mussakhojayeva, R. Yeshpanov, and H. A. Varol, “KazEmoTTS: A dataset for kazakh emotional text-to-speech synthesis,” *arXiv preprint arXiv:2404.01033*, 2024. [Online]. Available: <https://arxiv.org/abs/2404.01033>
- [31] R. Yeshpanov, “Using songs to improve kazakh automatic speech recognition,” *arXiv preprint arXiv:2603.00961*, 2026. [Online]. Available: <https://arxiv.org/abs/2603.00961>
- [32] InflexionLab, “InflexionLab/sybyrla · Hugging Face,” <https://huggingface.co/InflexionLab/sybyrla>, Mar. 2026, accessed: 2026-04-26.