



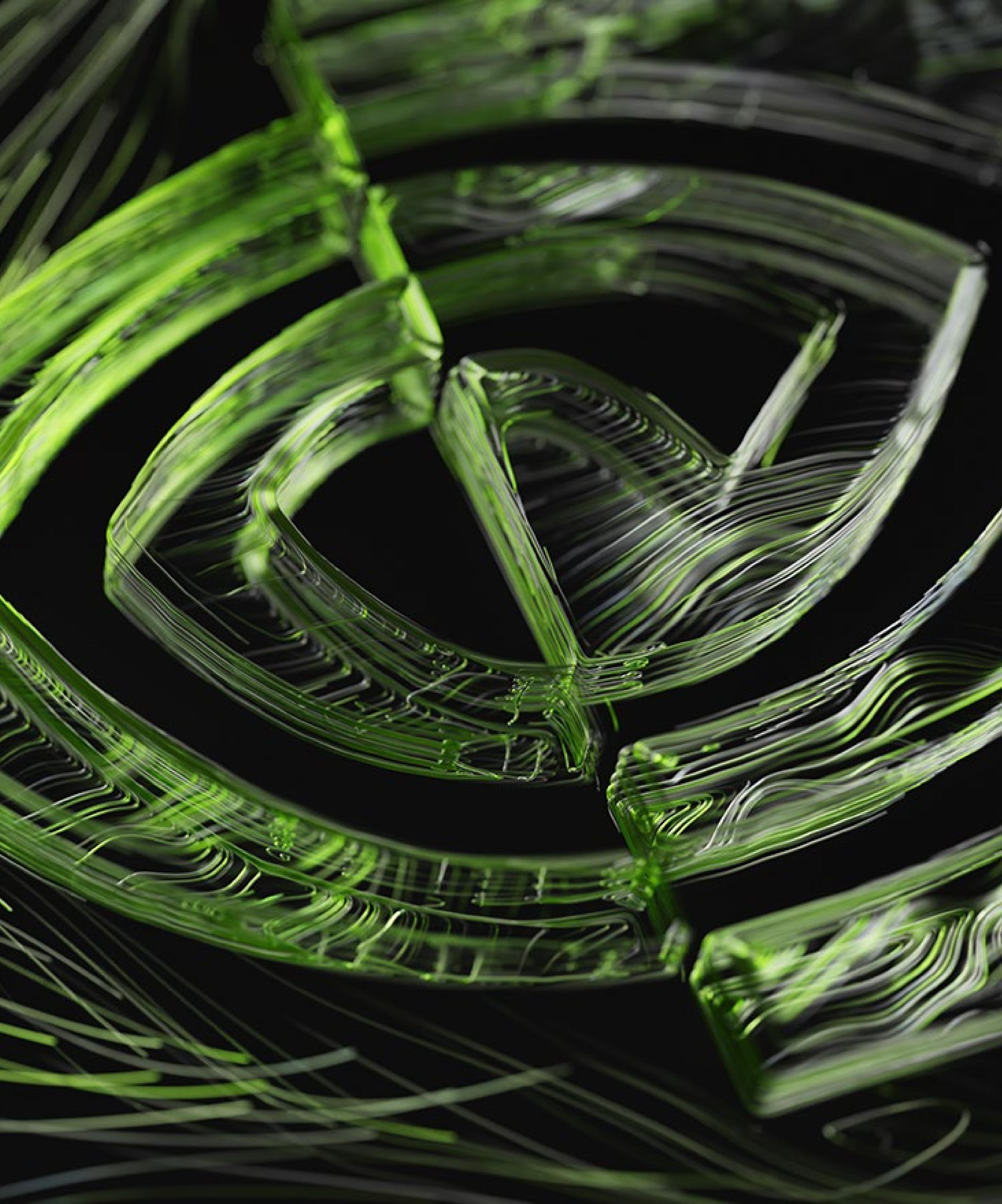
Part 2: TTS – Text To Speech and Conversational AI Customization

Building Conversational AI Applications



Course Timeline

8:50	—	9:00	Welcome
9:00	—	10:00	Lecture 1
10:00	—	11:00	Lab 1
11:00	—	11:15	Break
11:15	—	12:15	Lecture 2
12:15	—	13:15	Lab 2
13:15	—	14:00	break
14:00	—	15:00	Lecture 3
15:00	—	16:00	Lab 3
16:00	—	17:00	Assessment support



Course Agenda

Part 1: ASR - Automatic Speech Recognition

Lecture: Challenges of conversational AI applications, building blocks, ASR (automatic speech recognition) focus

Lab: Investigate ASR pipeline with NeMo and Riva

Part 2: TTS – Text To Speech and Conversational AI Customization

Lecture: How can we take advantage of conversational AI platforms to build custom applications?

Lab: Investigate the TTS pipeline with NeMo and Riva; then build a full custom conversational AI pipeline.

Part 3: Production Deployment

Lecture: Discussion of production deployment considerations including an overview of NVIDIA Triton Inference Server and TensorRT.

Lab: Hands-on deployment of an example application using Helm and Kubernetes; includes profiling the application



TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- **Speech Synthesis**
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

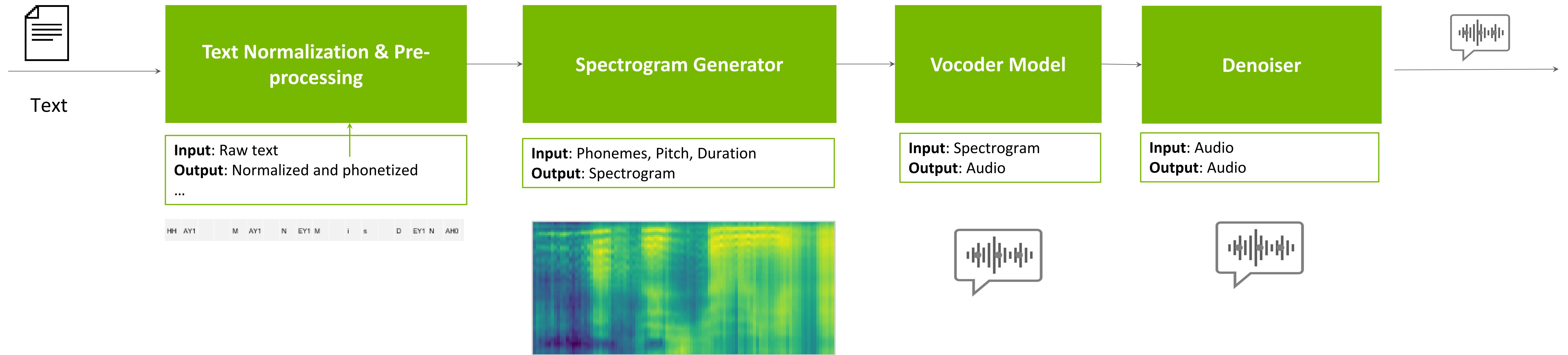
Part 2: Lecture

- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Part 2: Lab

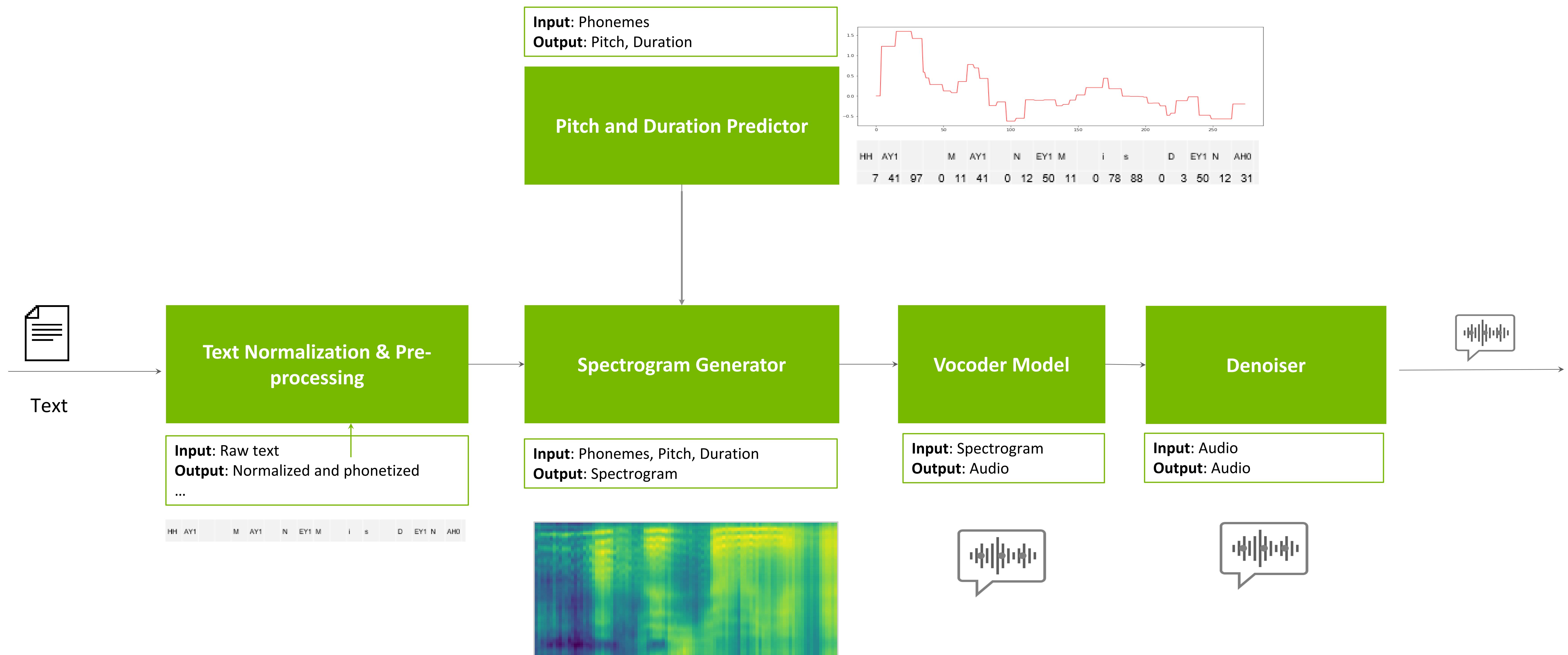
- ASR, TTS and NLP Customization

TTS Pipeline



TTS Pipeline

Expressiveness | Multi-Speakers | Style Transfer



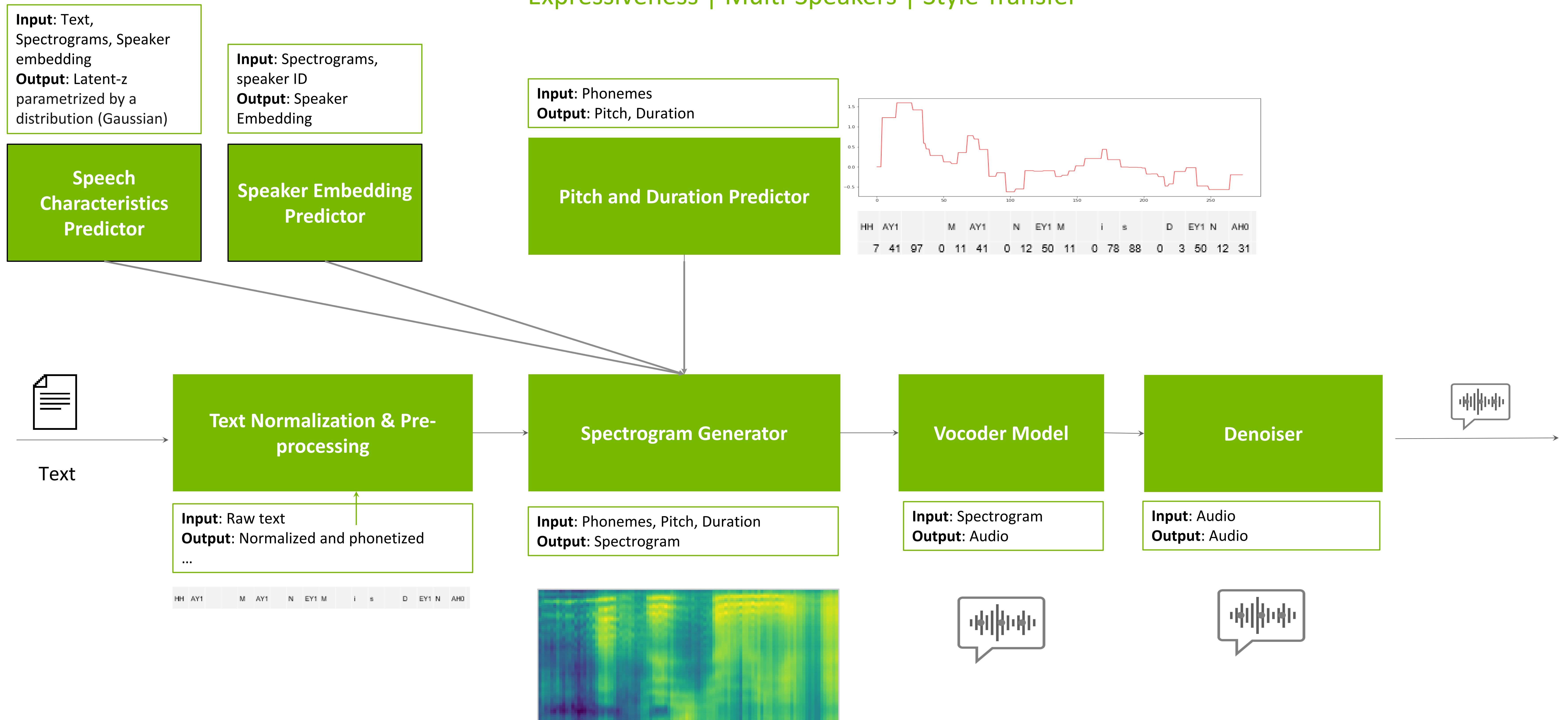
TTS Pipeline

Expressiveness | Multi-Speakers | Style Transfer

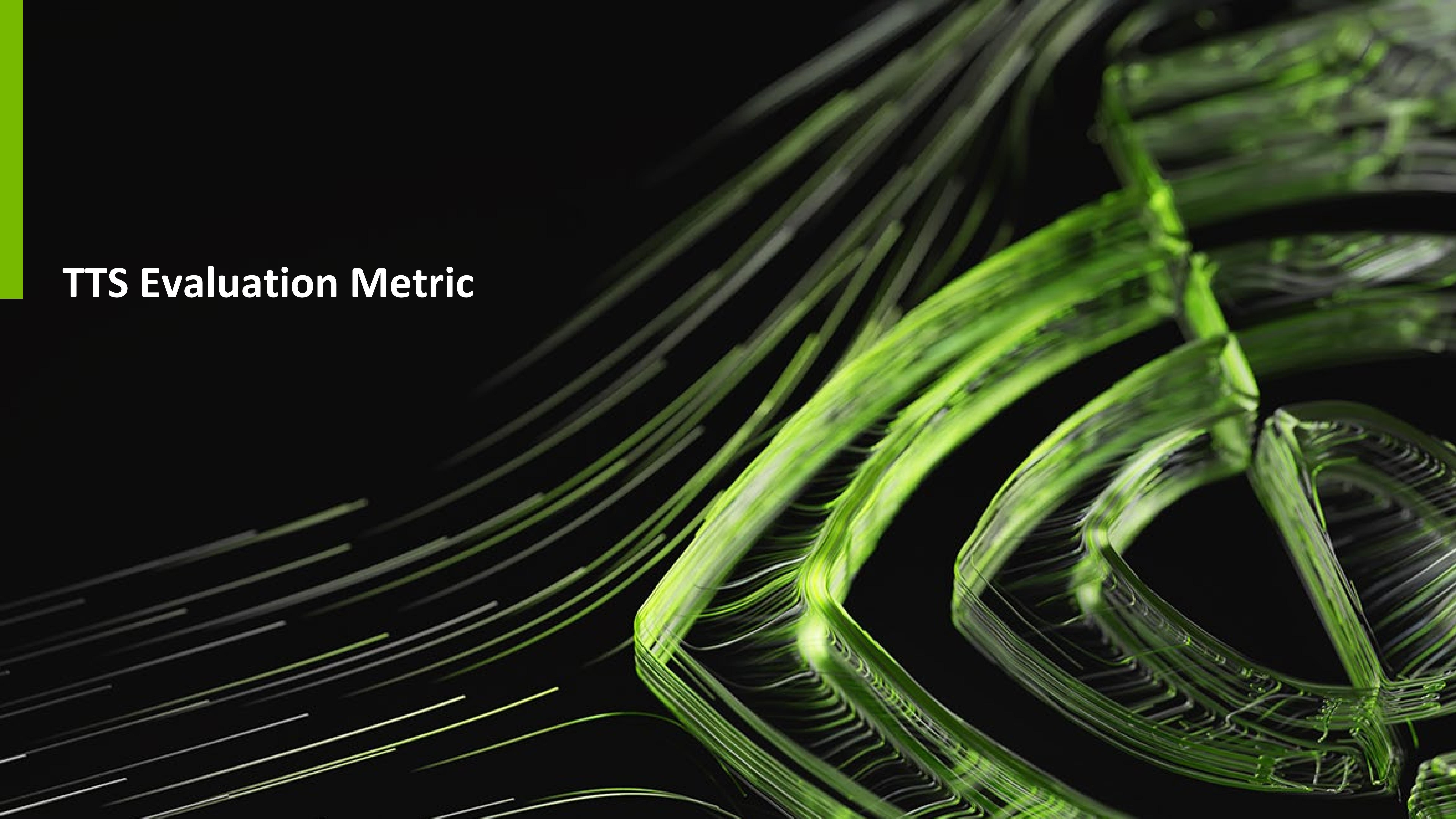


TTS Pipeline

Expressiveness | Multi-Speakers | Style Transfer



TTS Evaluation Metric



Measuring Success of Speech Synthesis

Mean opinion score

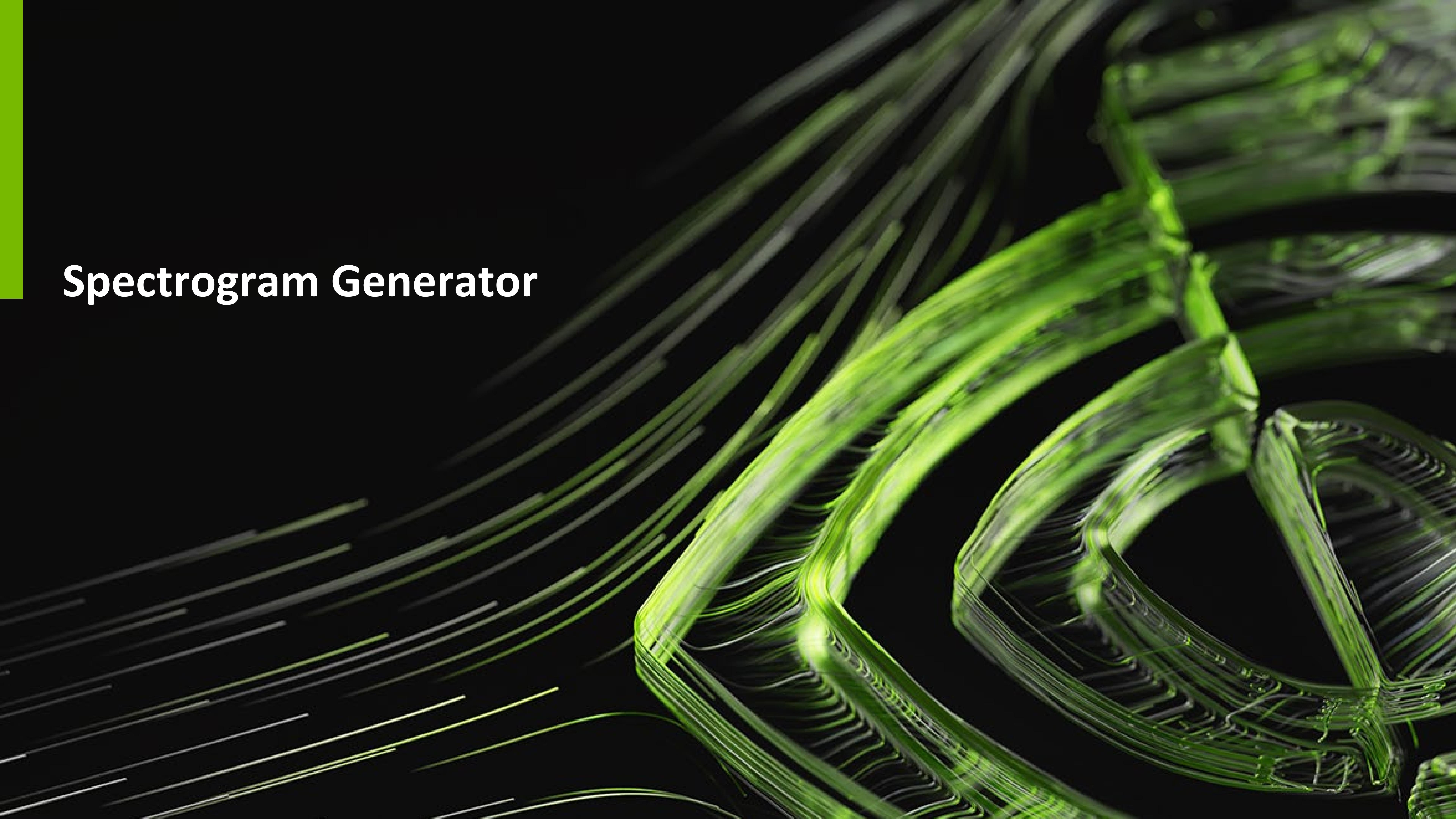
It is the arithmetic mean over all individual "values on a predefined scale that a subject assigns to his opinion of the performance of a system quality".

$$MOS = \frac{\sum_{n=1}^N R_n}{N}$$

Where R are the individual ratings for a given stimulus by N subjects.

Rating	Label
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

Spectrogram Generator



Spectrogram Generator

Deep Voice 3, Tacotron, Tacotron 2 | Autoregressive Models

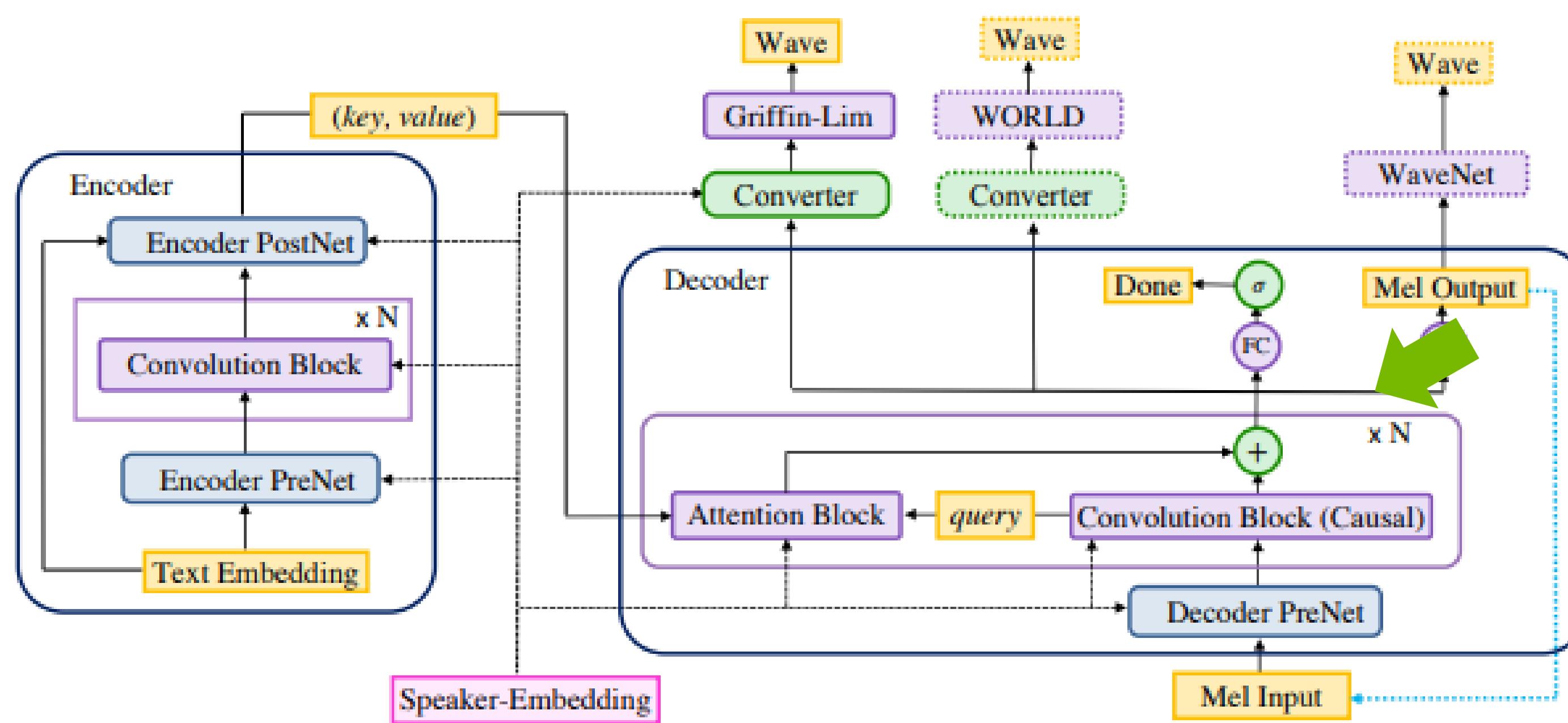
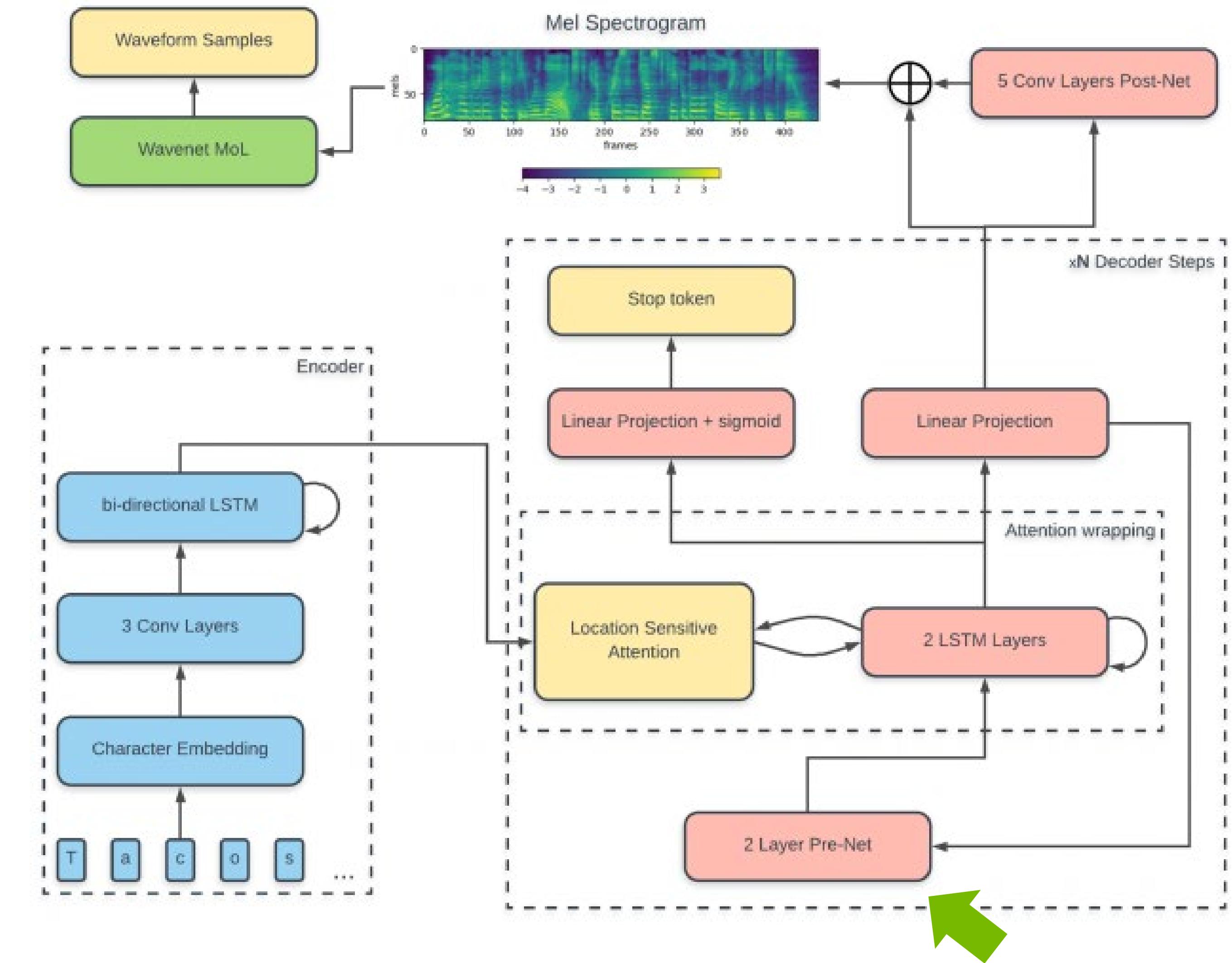


Figure 1: Deep Voice 3 uses residual convolutional layers to encode text into per-timestep *key* and *value* vectors for an attention-based decoder. The decoder uses these to predict the mel-scale log magnitude spectrograms that correspond to the output audio. (Light blue dotted arrows depict the autoregressive process during inference.) The hidden states of the decoder are then fed to a converter network to predict the vocoder parameters for waveform synthesis. See Appendix A for more details.



Model	MOS (VCTK)	MOS (LibriSpeech)
Deep Voice 3 (Griffin-Lim)	3.01 ± 0.29	2.37 ± 0.24
Deep Voice 3 (WORLD)	3.44 ± 0.32	2.89 ± 0.38
Deep Voice 2 (WaveNet)	3.69 ± 0.23	-
Tacotron (Griffin-Lim)	2.07 ± 0.31	-
Ground truth	4.69 ± 0.04	4.51 ± 0.18

System	MOS
Parametric	3.492 ± 0.096
Tacotron (Griffin-Lim)	4.001 ± 0.087
Concatenative	4.166 ± 0.091
WaveNet (Linguistic)	4.341 ± 0.051
Ground truth	4.582 ± 0.053
Tacotron 2 (this paper)	4.526 ± 0.066

Table 1. Mean Opinion Score (MOS) evaluations with 95% confidence intervals computed from the t-distribution for various systems.

Spectrogram Generator

FASTSPEECH, FastPitch - Condition to duration and pitch | Parallel TTS

FastPitch:

- Train Pitch, duration and spectrograms predictors.
- Ground truth p and d are used during training,
- Loss mean-squared error (MSE) between the predicted spectrogram, pitch and duration

$$\mathcal{L} = \|\hat{\mathbf{y}} - \mathbf{y}\|_2^2 + \alpha\|\hat{\mathbf{p}} - \mathbf{p}\|_2^2 + \gamma\|\hat{\mathbf{d}} - \mathbf{d}\|_2^2.$$

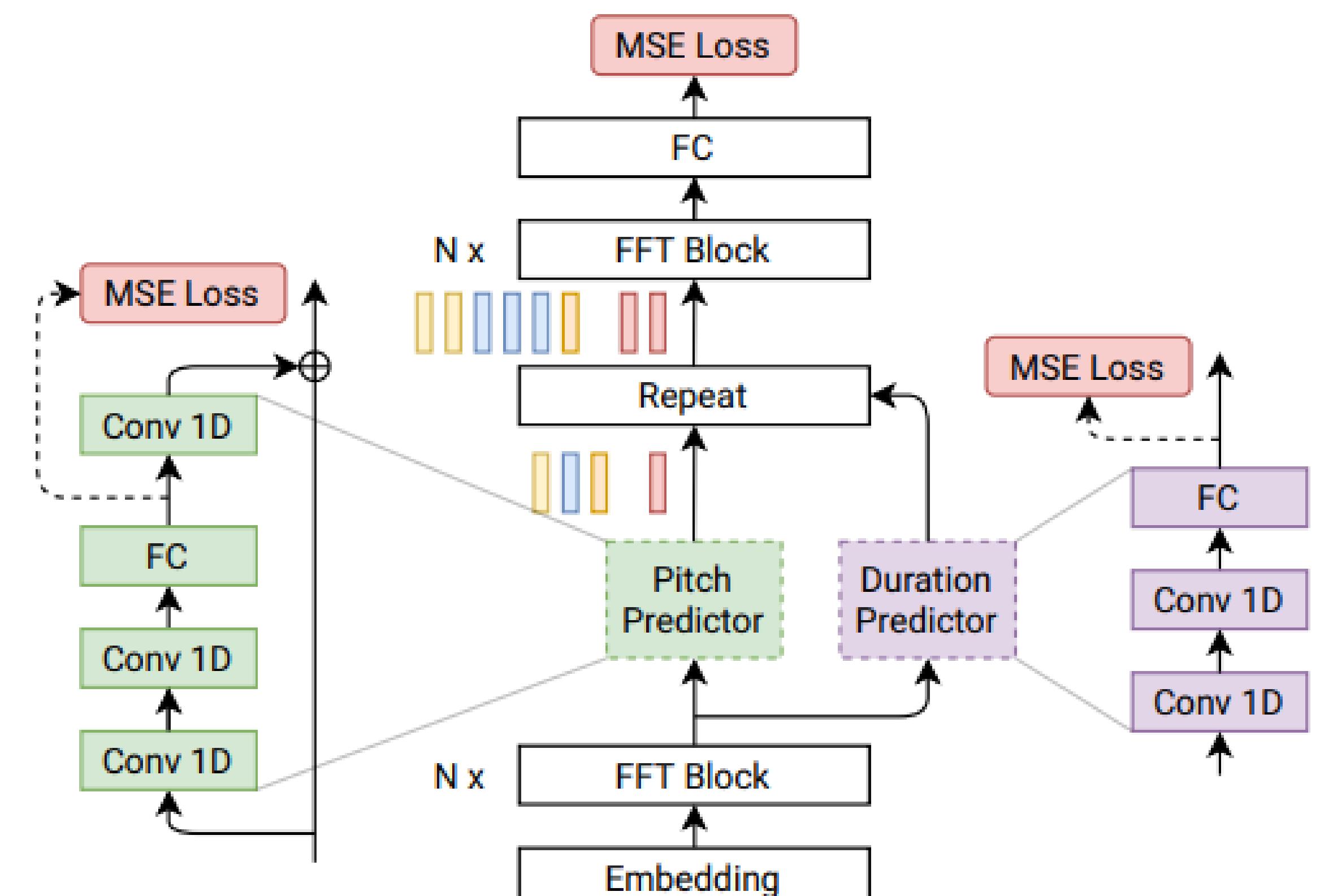


Fig. 1. Architecture of FastPitch follows FastSpeech [1]. A single pitch value is predicted for every temporal location.

Table 2. Multi-speaker Mean Opinion Scores with 95% confidence intervals, evaluated on the LJSpeech dev samples

Model	MOS
Tacotron 2 (Mel + WaveGlow)	3.707 ± 0.218
Flowtron (Mel + WaveGlow)	3.882 ± 0.159
FastPitch (Mel + WaveGlow)	4.071 ± 0.164

Table 3. Average latency and real-time factor of mel-spectrogram generation on the first 2048 utterances from LJSpeech training subset

Model	Latency	Mel RTF
Tacotron 2 (GPU)	0.4109 ± 0.2631 s	$15.42 \times$
FastPitch (CPU)	0.0602 ± 0.0205 s	$107.59 \times$
FastPitch (GPU)	0.0071 ± 0.0010 s	$911.86 \times$

- Listen to samples:
 - Fastspeech2: <https://speechresearch.github.io/fastspeech2/>
 - FastPitch: <https://fastpitch.github.io/>

Spectrogram Generator

Glow TTS - Transformers | Parallel TTS

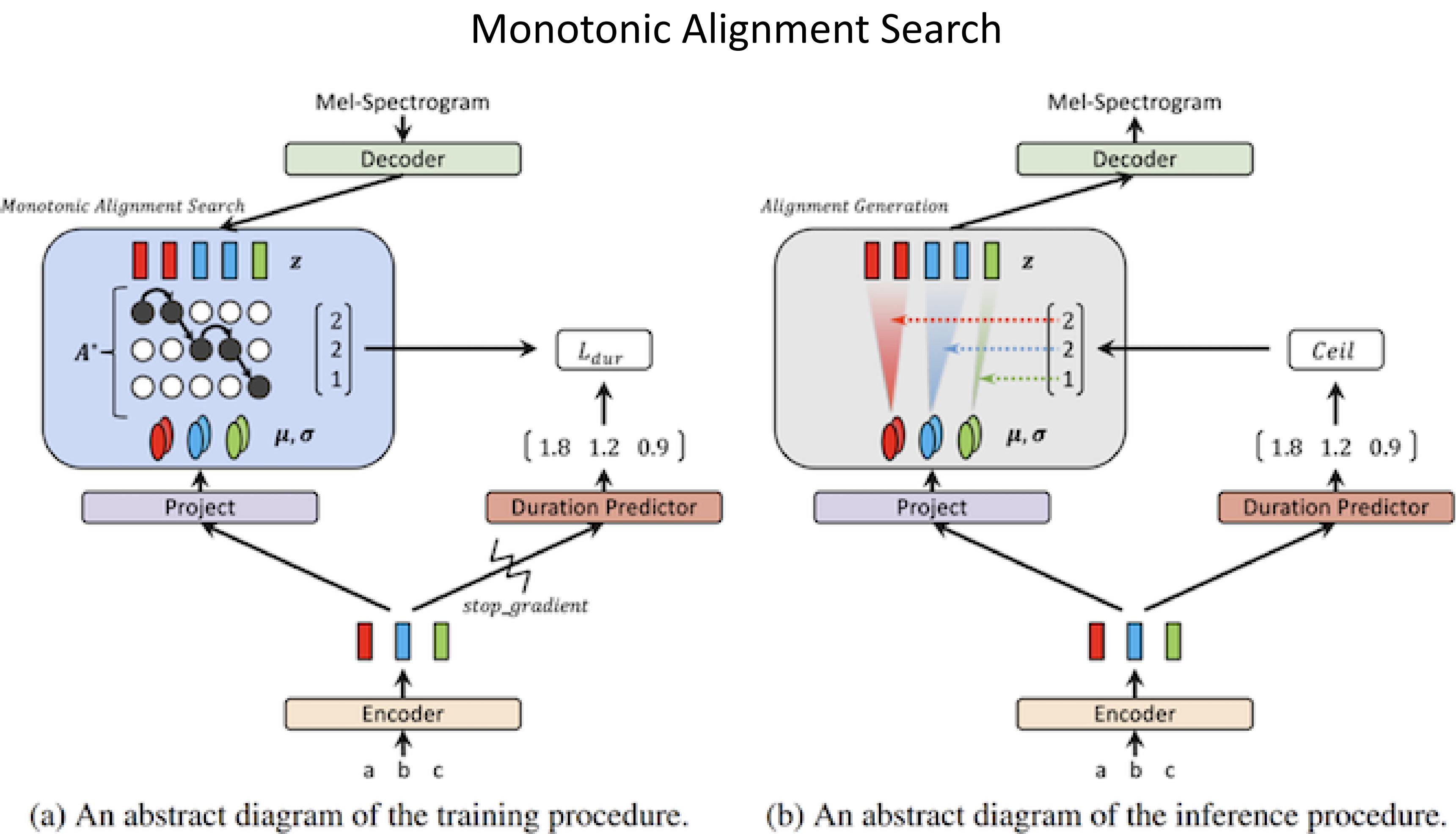
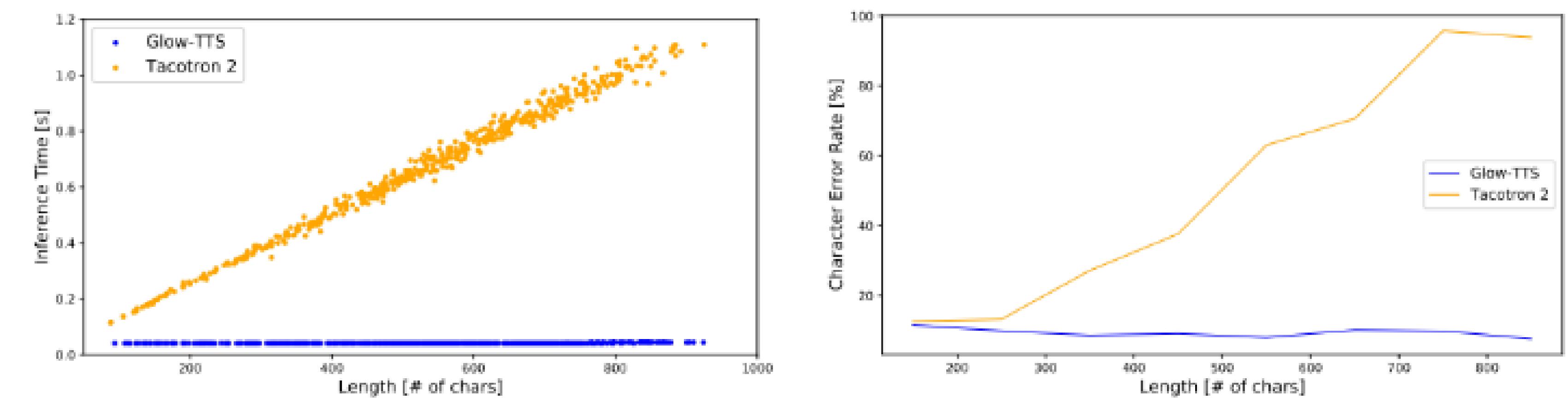


Figure 1: Training and inference procedures of Glow-TTS.

Table 2: The Mean Opinion Score (MOS) of a multi-speaker TTS with 95% confidence intervals.

Method	9-scale MOS
GT	4.54 ± 0.07
GT (Mel + WaveGlow)	4.22 ± 0.07
Tacotron2 (Mel + WaveGlow)	3.35 ± 0.12
Glow-TTS ($T = 0.333$, Mel + WaveGlow)	3.20 ± 0.12
Glow-TTS ($T = 0.500$, Mel + WaveGlow)	3.31 ± 0.12
Glow-TTS ($T = 0.667$, Mel + WaveGlow)	3.45 ± 0.11

"Glow-TTS can generate mel-spectrograms **15.7 times faster** than the autoregressive TTS model, Tacotron 2, while obtaining comparable performance."



(a) The inference time comparison for Tacotron 2 and Glow-TTS (yellow: Tacotron2, blue: Glow-TTS).

(b) Robustness to the length of input utterances (yellow: Tacotron2, blue: Glow-TTS).

Figure 3: Comparison of inference time and length robustness.

- No external aligner, learn its own alignment using Monotonic Alignment Search
- Listen to Glow-TTS samples: <https://jaywalnut310.github.io/glow-tts-demo/index.html>



Spectrogram Generator

Diversity | RadTTS

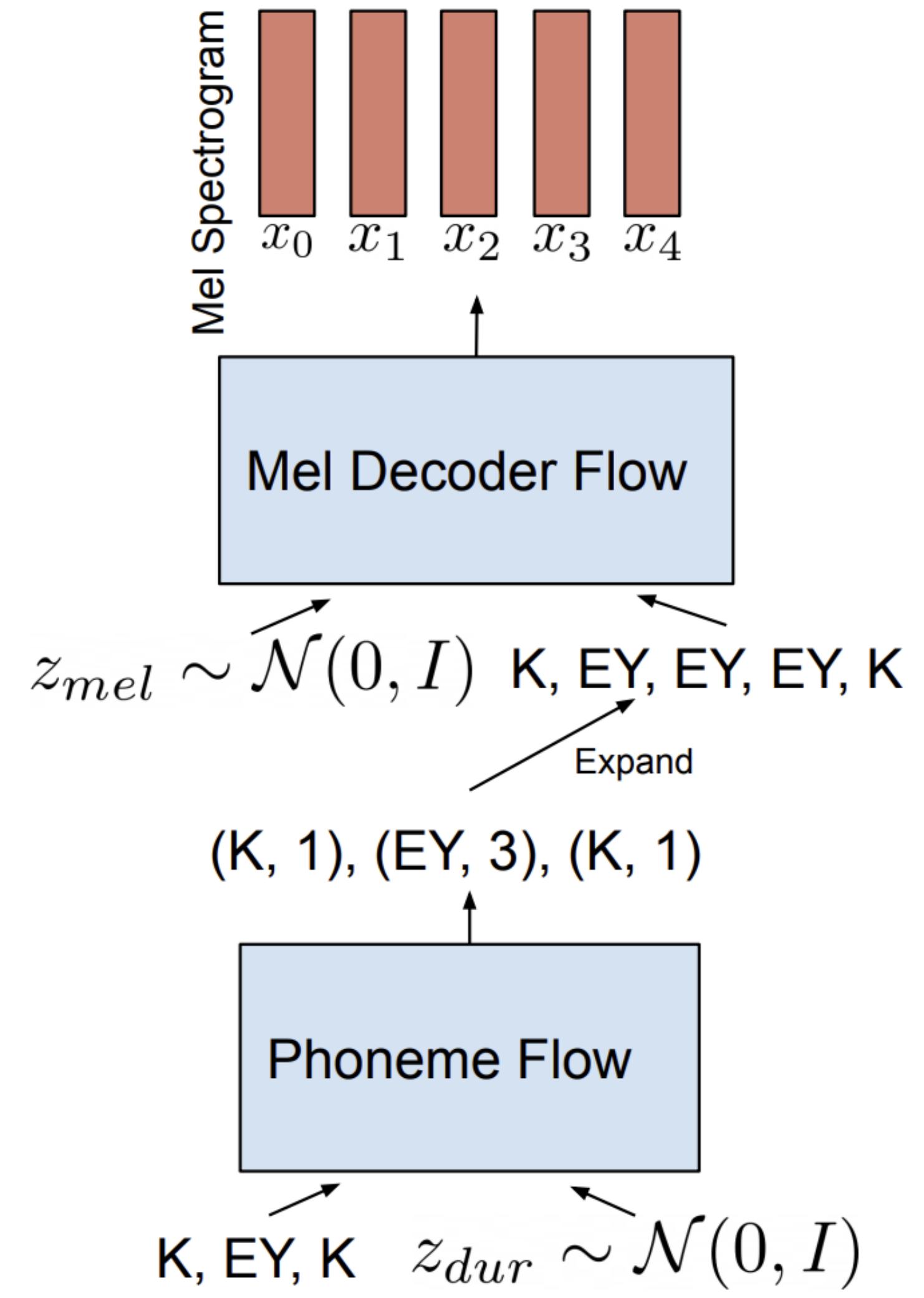
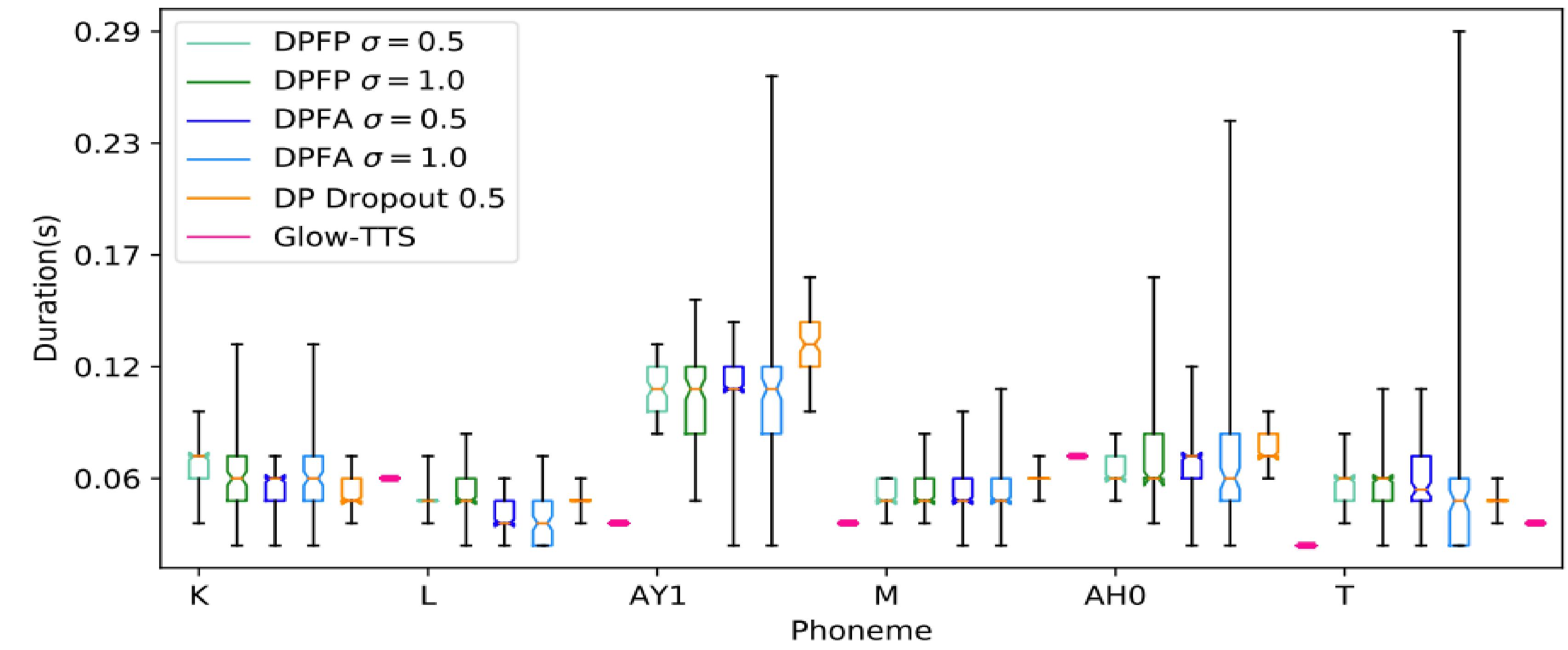


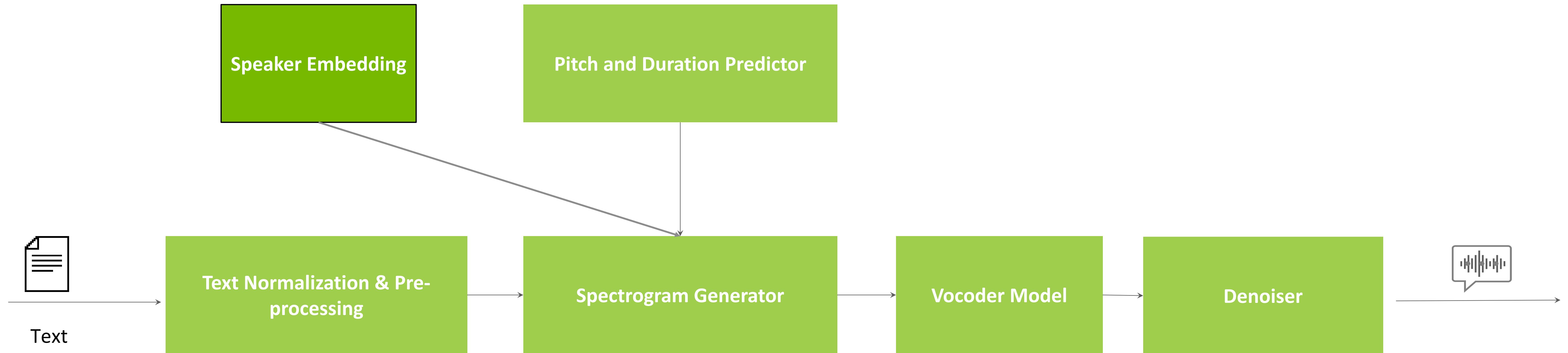
Figure 1. Simplified inference pipeline for the proposed model. The phoneme flow first samples from $P_{dur}()$ to attain per-phoneme durations (\mathcal{A}), which are then used to prepare the input to the parallel Mel-Decoder flow that models $P_{mel}()$



Evaluation of RAD-TTS wrt other non-autoregressive models in terms of variability in speech.

Spectrogram Generator

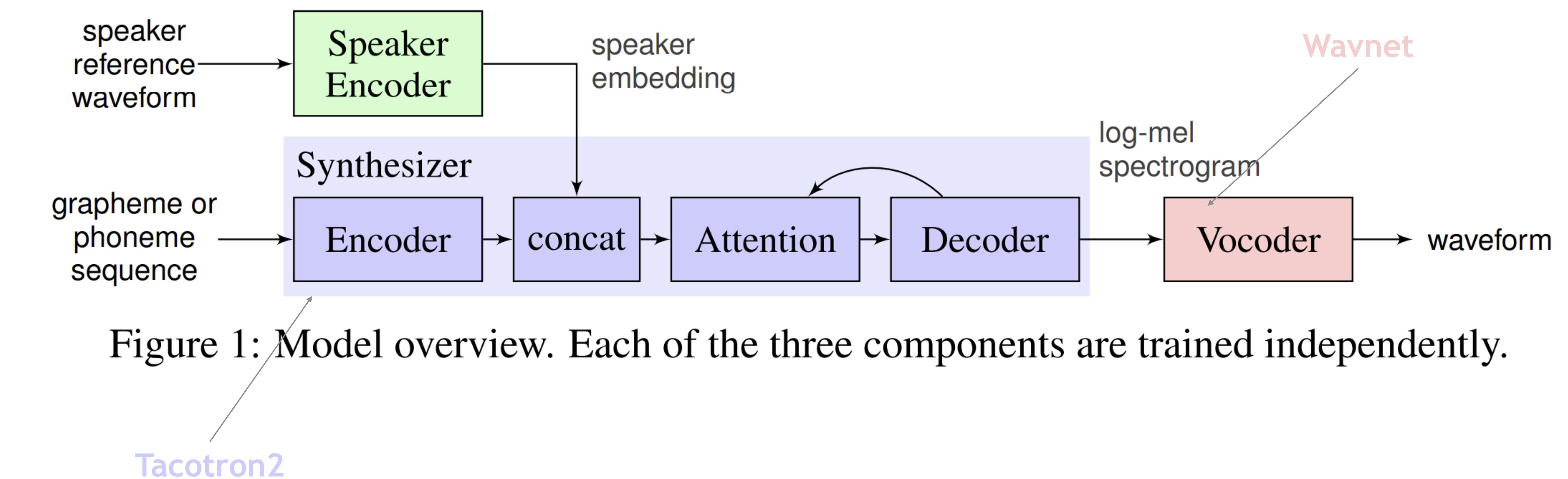
Multi-Speakers TTS | Spectrogram Generator conditioned to Speaker



Spectrogram Generator

Multi-Speakers TTS | Spectrogram Generator conditioned to Speaker

- Speaker encoder, Spectrogram Generator and Vocoders are trained independently
- Speaker Encoder:
 - **Input:** Spectrograms
 - **Network:** Stack of 3 LSTM layers, each followed by a projection to 256 dimensions
 - **Output:** Averaged and normalized embedding on sliding window
 - Trained for text-independent speaker verification task.



Spectrogram Generator

Multi-Speakers TTS | Spectrogram Generator conditioned to Speaker

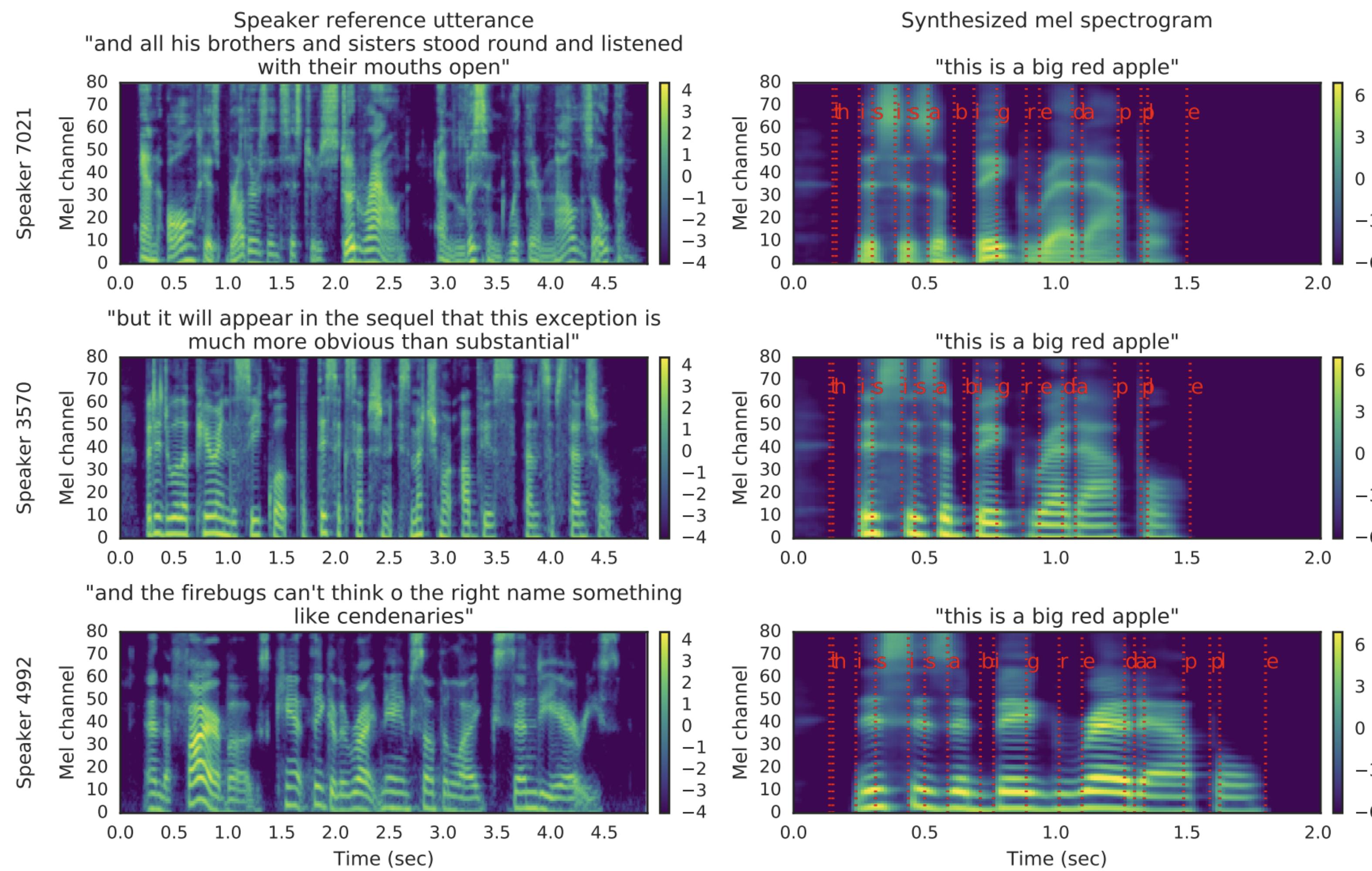


Figure 2: Example synthesis of a sentence in different voices using the proposed system. Mel spectrograms are visualized for reference utterances used to generate speaker embeddings (left), and the corresponding synthesizer outputs (right). The text-to-spectrogram alignment is shown in red. Three speakers held out of the train sets are used: one male (top) and two female (center and bottom).

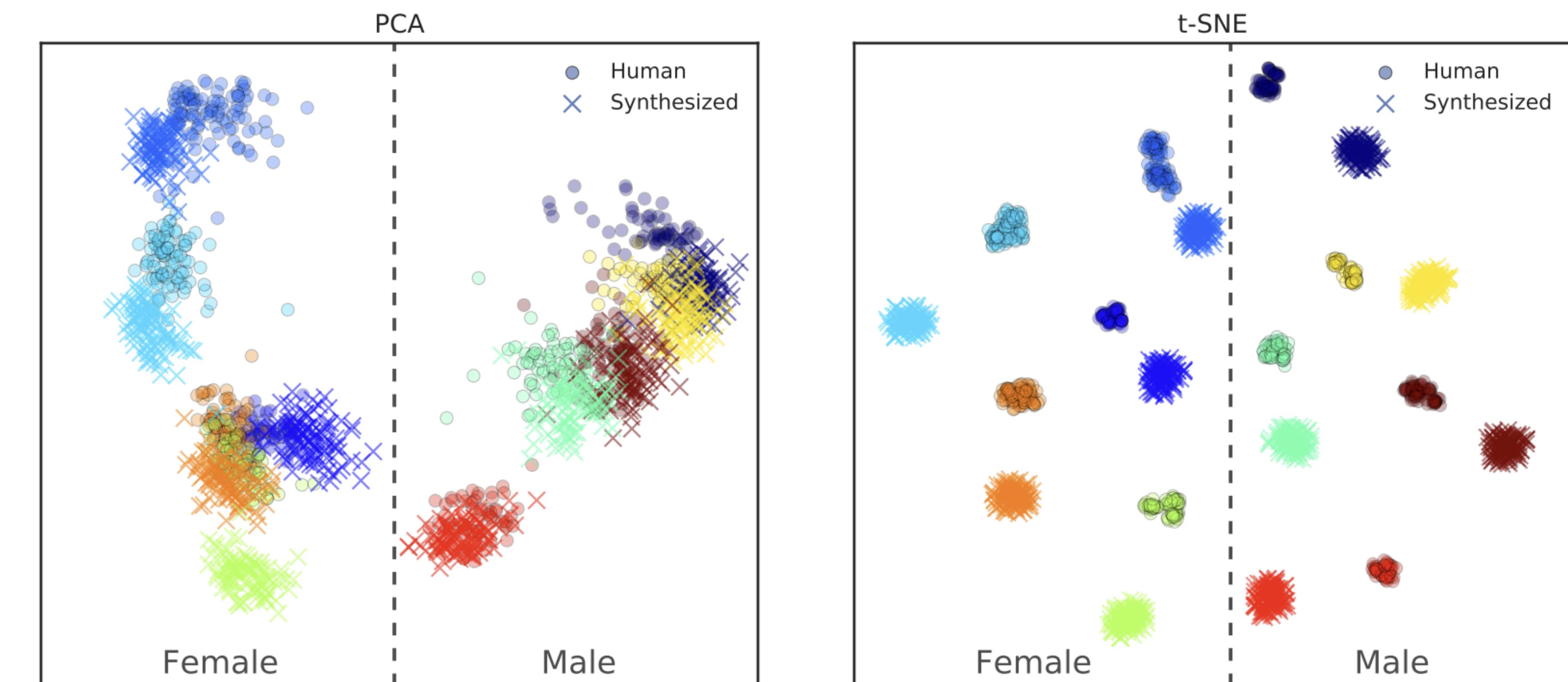
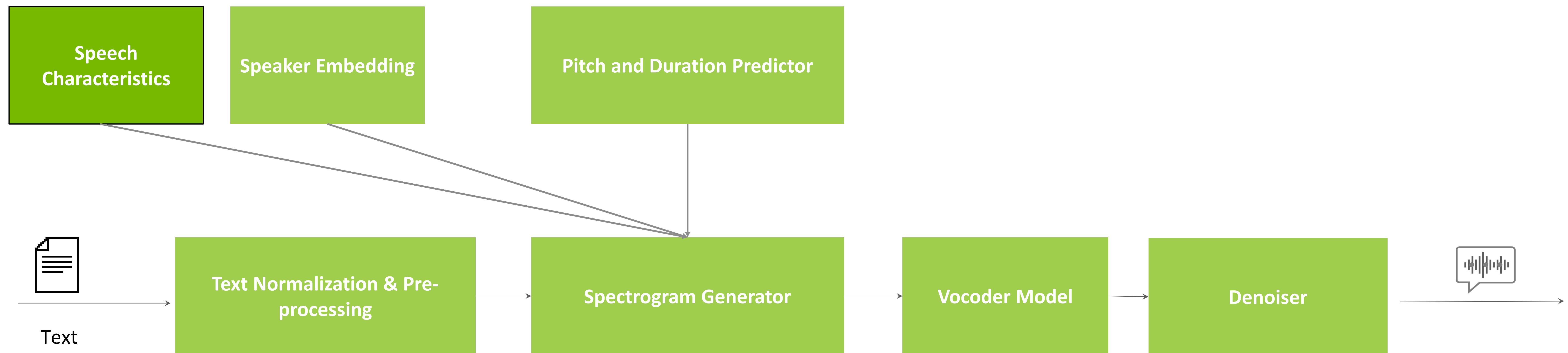


Figure 3: Visualization of speaker embeddings extracted from LibriSpeech utterances. Each color corresponds to a different speaker. Real and synthetic utterances appear nearby when they are from the same speaker, however real and synthetic utterances consistently form distinct clusters.

Spectrogram Generator

Style Transfer | Spectrogram Generator conditioned to a style



Spectrogram Generator

Style Transfer | FlowTron

- **Speech characteristics:** Maps a distribution over mel spectrograms and text to a latent z-space parametrized by a simple distribution (Gaussian).
- Flowtron can transfer the style from data or speakers not seen in the training
- **At inference:** Either sample from all speaking styles or sample from specific speaking styles.
 - Provide audio samples to find the region in z-space that is associated with its specific speaking style (red dots)
 - Compute their centroid
 - Use this centroid as the mean of a Gaussian distribution (red circles) and adjust the variance one more time to adjust the span of the region in the z-space being sampled

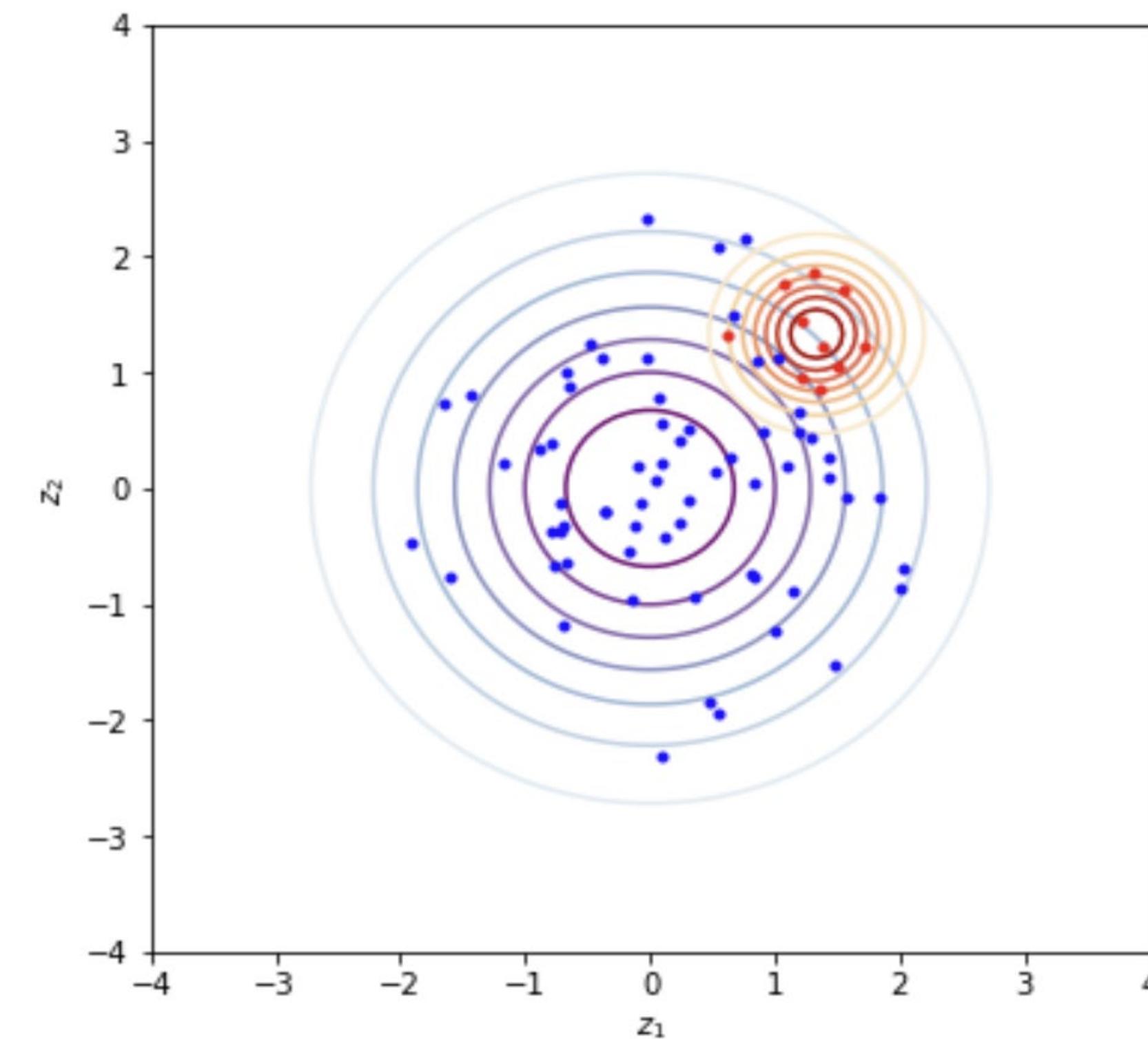


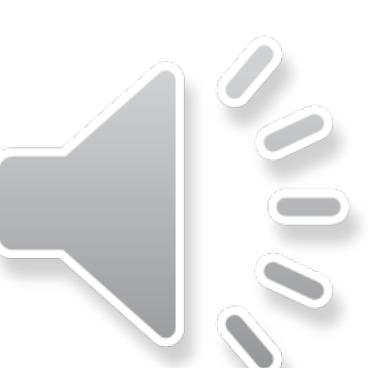
Figure 2. Illustration of a Gaussian distribution and Flowtron's region in z-space associated with the style present in audio samples. Hue represents how often that region will be sampled.



Baseline (sally)



Style (surprised)



Target (Sally sounds surprised)

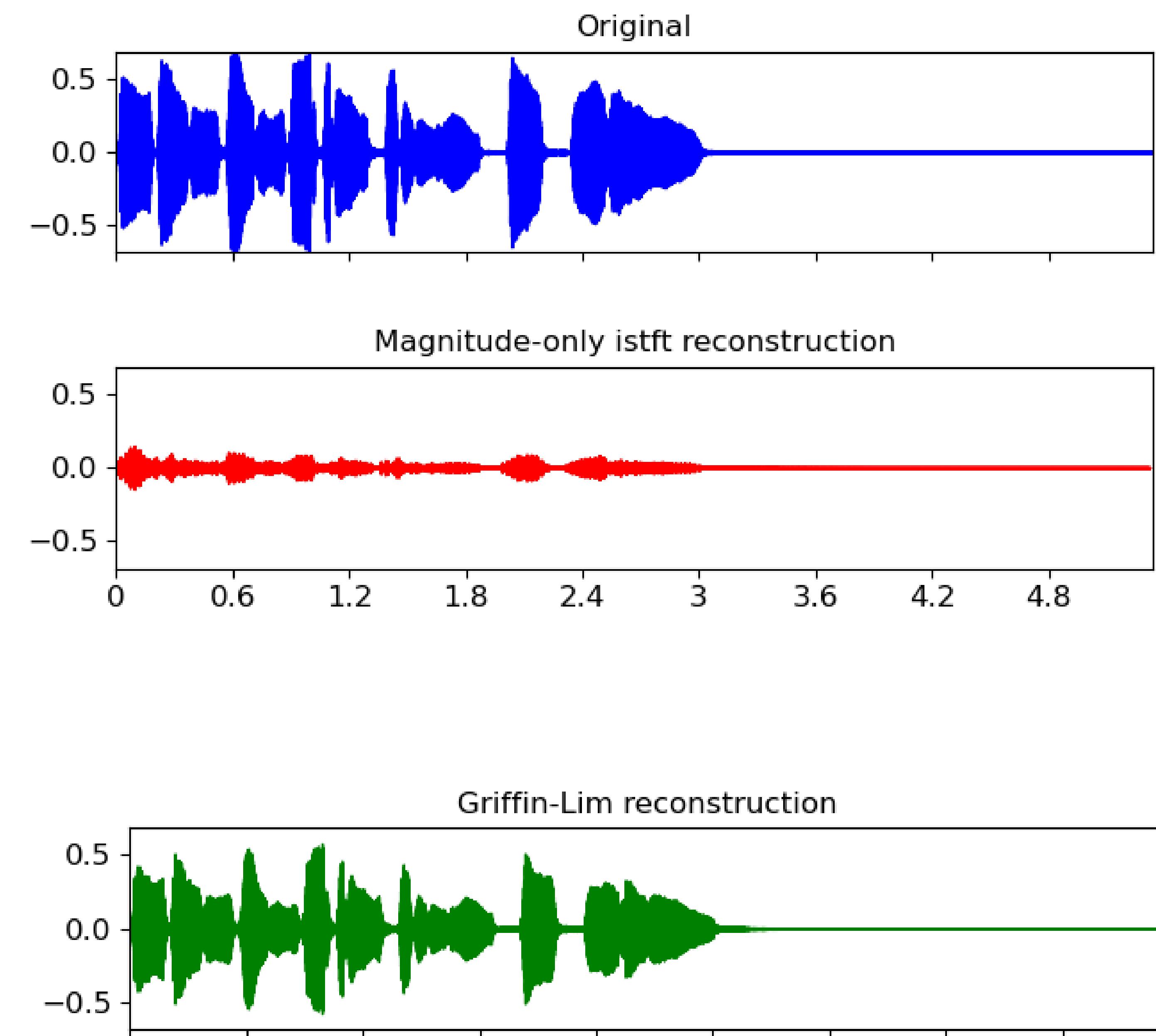
Audio Generation (Vocoder)



Vocoder

Griffin-Lim Algorithm

- Why not use Inverse Fourier Transform to get back to the temporal domain (wave format) from the synthesized spectrograms?
- With the spectrograms we have the amplitude, but we lost the phase
- Griffin-Lim algorithm:
 - Given a Short-Time Fourier Transform magnitude matrix (S), the algorithm randomly initializes phase estimates, and then alternates forward- and inverse-STFT operations until satisfying result



WaveGlow, SqueezeWave

A Non autoregression combination of WaveNet + Glow

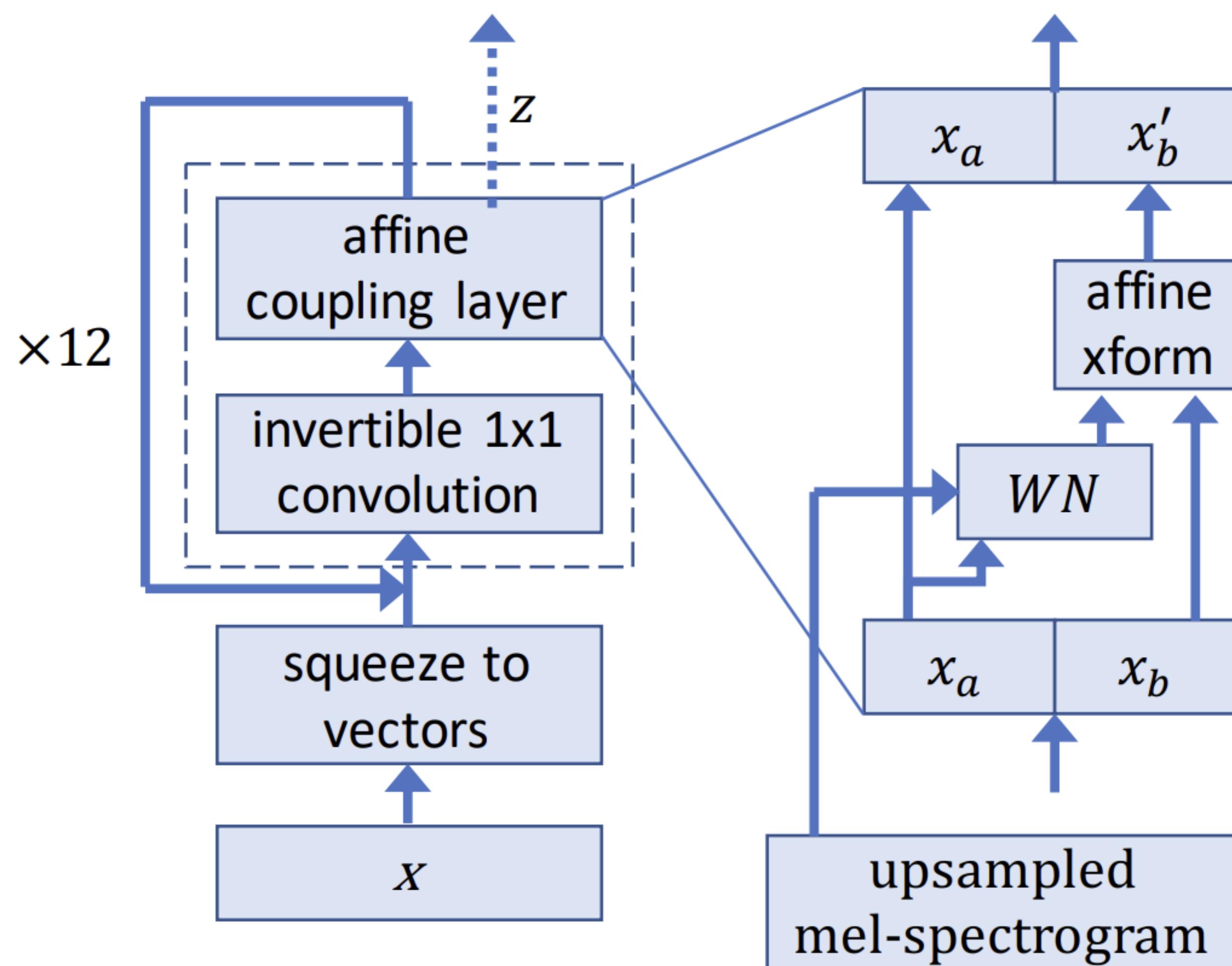


Fig. 1: WaveGlow network

Model	Mean Opinion Score (MOS)
Griffin-Lim	3.823 ± 0.1349
WaveNet	3.885 ± 0.1238
WaveGlow	3.961 ± 0.1343
Ground Truth	4.274 ± 0.1340

Table 1: Mean Opinion Scores

HiFiGan

Matching performance of autoregressive models

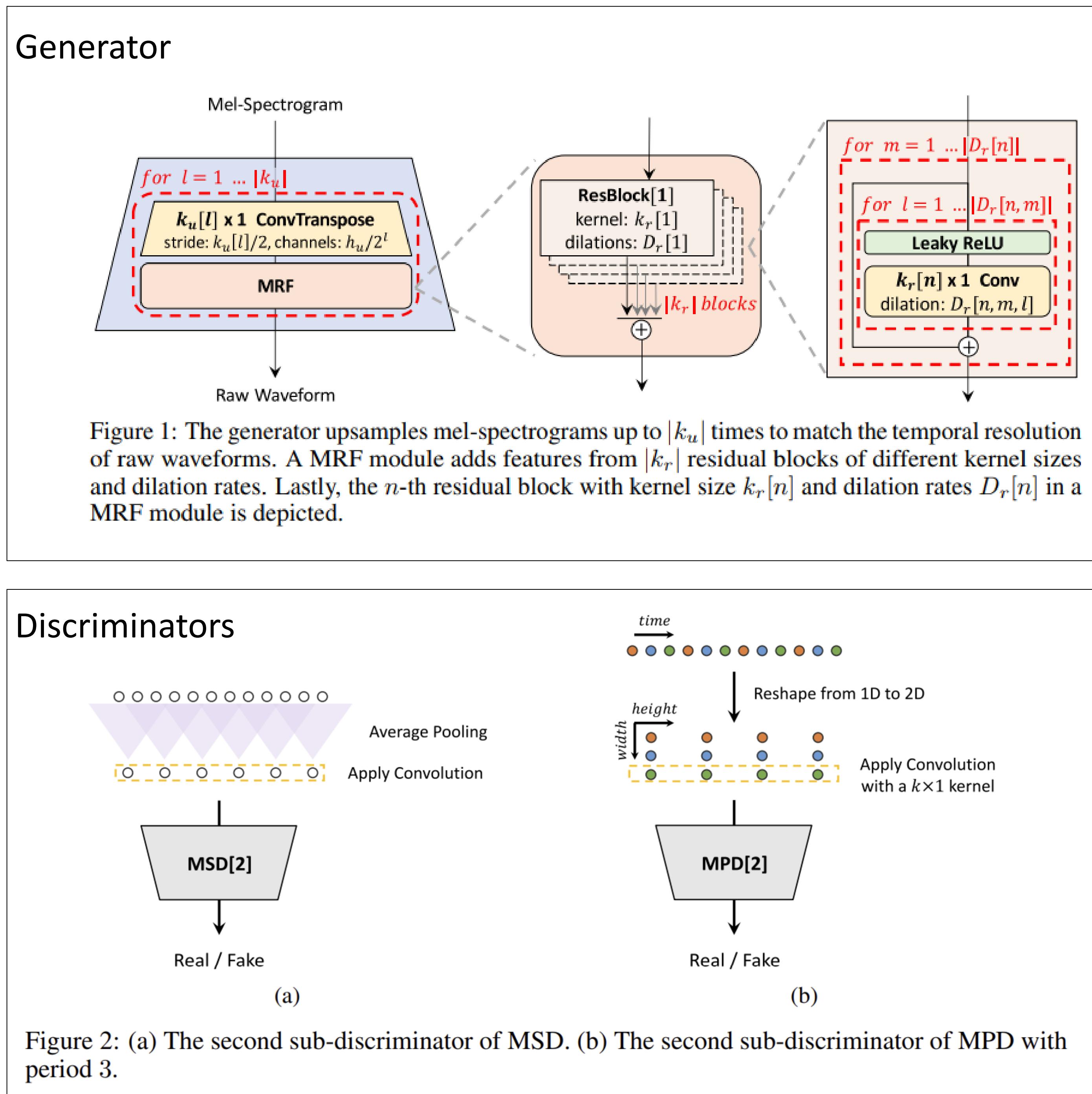


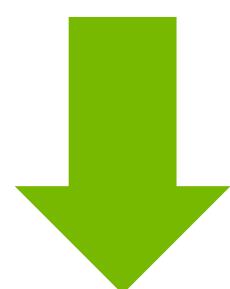
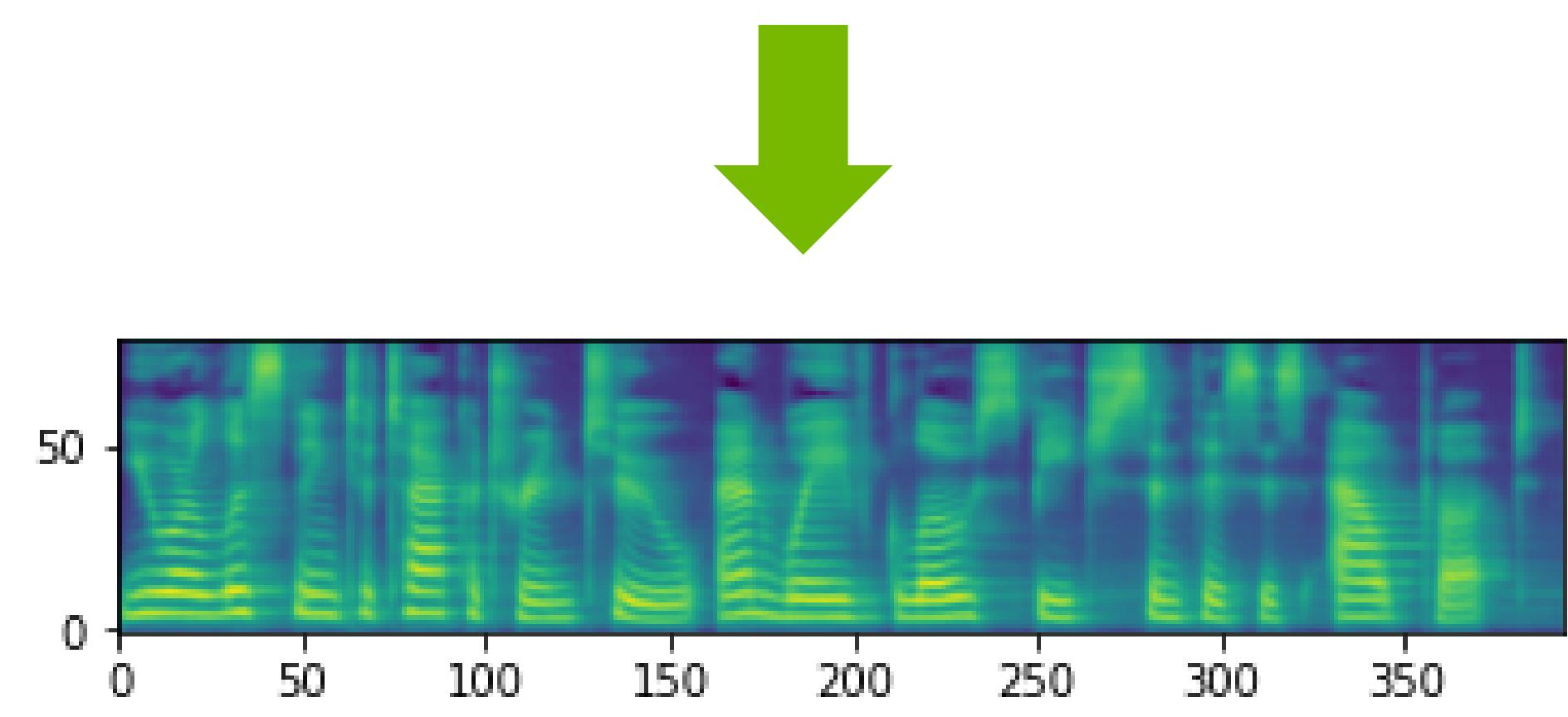
Table 1: Comparison of the MOS and the synthesis speed. Speed of n kHz means that the model can generate $n \times 1000$ raw audio samples per second. The numbers in () mean the speed compared to real-time.

Model	MOS (CI)	Speed on CPU (kHz)	Speed on GPU (kHz)	# Param (M)
Ground Truth	4.45 (± 0.06)	—	—	—
WaveNet (MoL)	4.02 (± 0.08)	—	0.07 ($\times 0.003$)	24.73
WaveGlow	3.81 (± 0.08)	4.72 ($\times 0.21$)	501 ($\times 22.75$)	87.73
MelGAN	3.79 (± 0.09)	145.52 ($\times 6.59$)	14,238 ($\times 645.73$)	4.26
HiFi-GAN V1	4.36 (± 0.07)	31.74 ($\times 1.43$)	3,701 ($\times 167.86$)	13.92
HiFi-GAN V2	4.23 (± 0.07)	214.97 ($\times 9.74$)	16,863 ($\times 764.80$)	0.92
HiFi-GAN V3	4.05 (± 0.08)	296.38 ($\times 13.44$)	26,169 ($\times 1,186.80$)	1.46

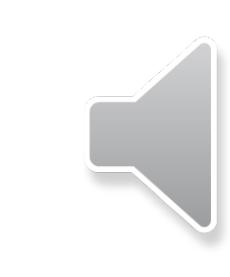
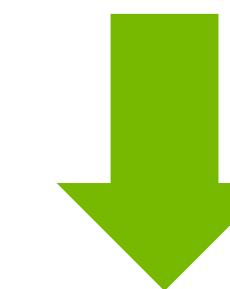
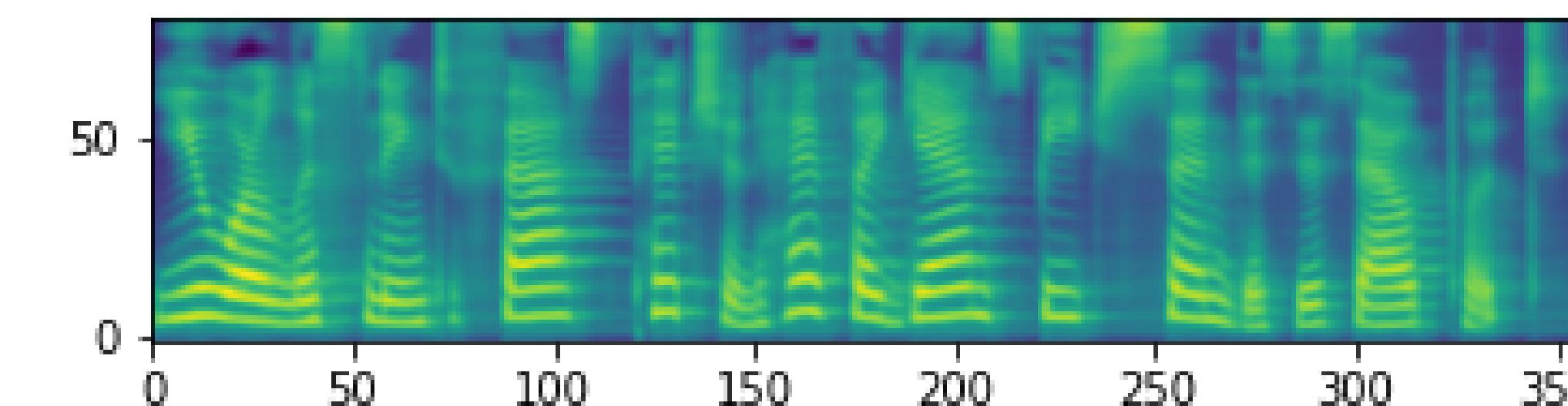
Text-To-Speech (TTS)

Deep learning based models

*You are listening to Tacotron two and
wave glow speech synthesis output*



*You are listening to Fast Pitch and
Hifi GAN speech synthesis output*



Try it: https://github.com/NVIDIA/NeMo/tutorials/tts/Inference_ModelSelect.ipynb

End-to-End Speech Synthesis



End-to-End Synthesis

WaveNet - autoregressive

Apply over an area larger than its length by skipping input values with a certain step

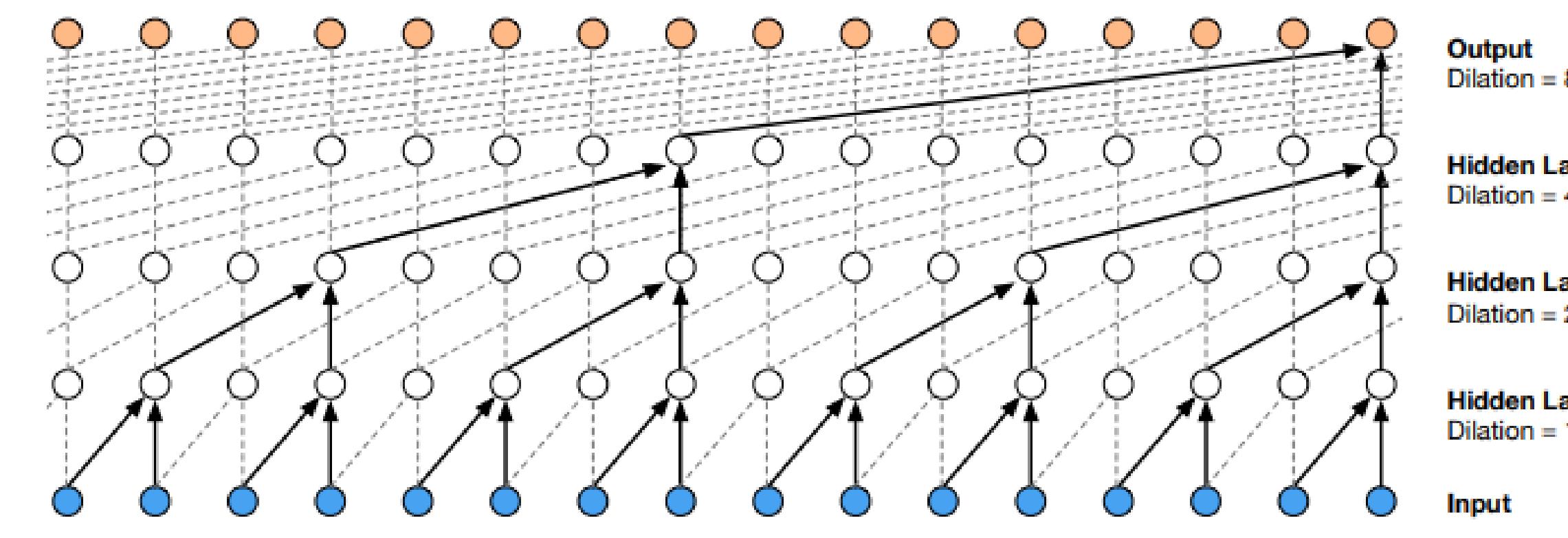


Figure 3: Visualization of a stack of *dilated* causal convolutional layers.

Predictions at timesteps t cannot depend on the future timesteps

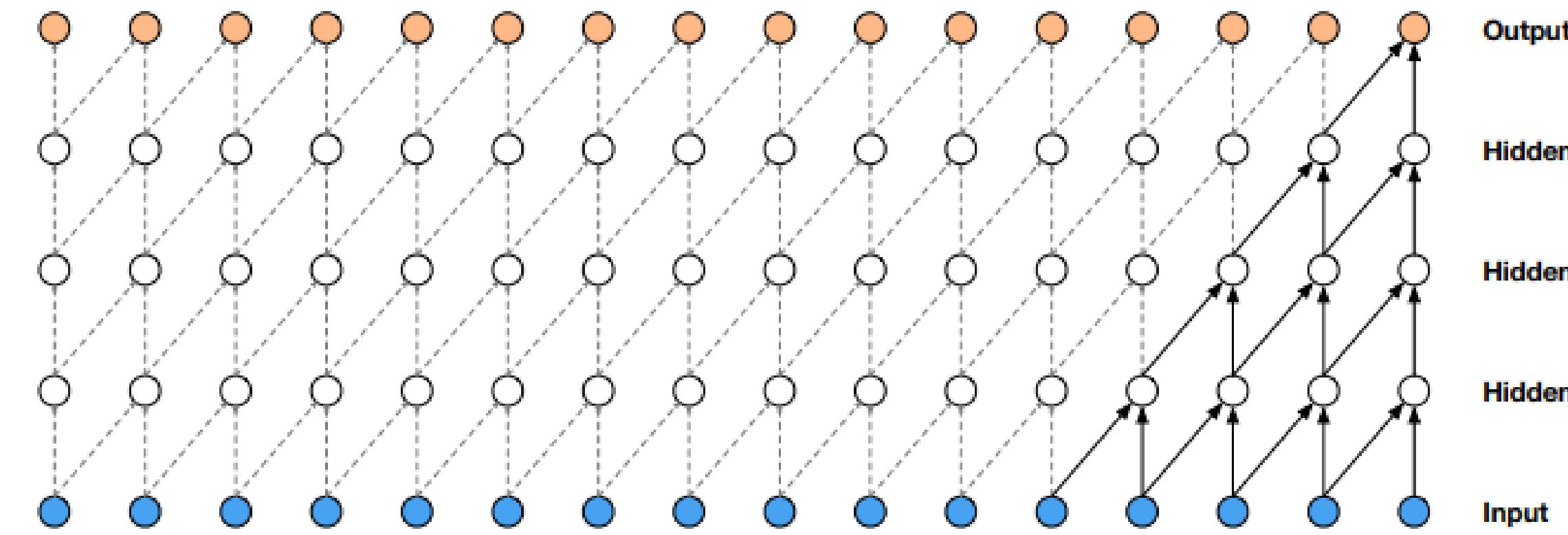


Figure 2: Visualization of a stack of causal convolutional layers.

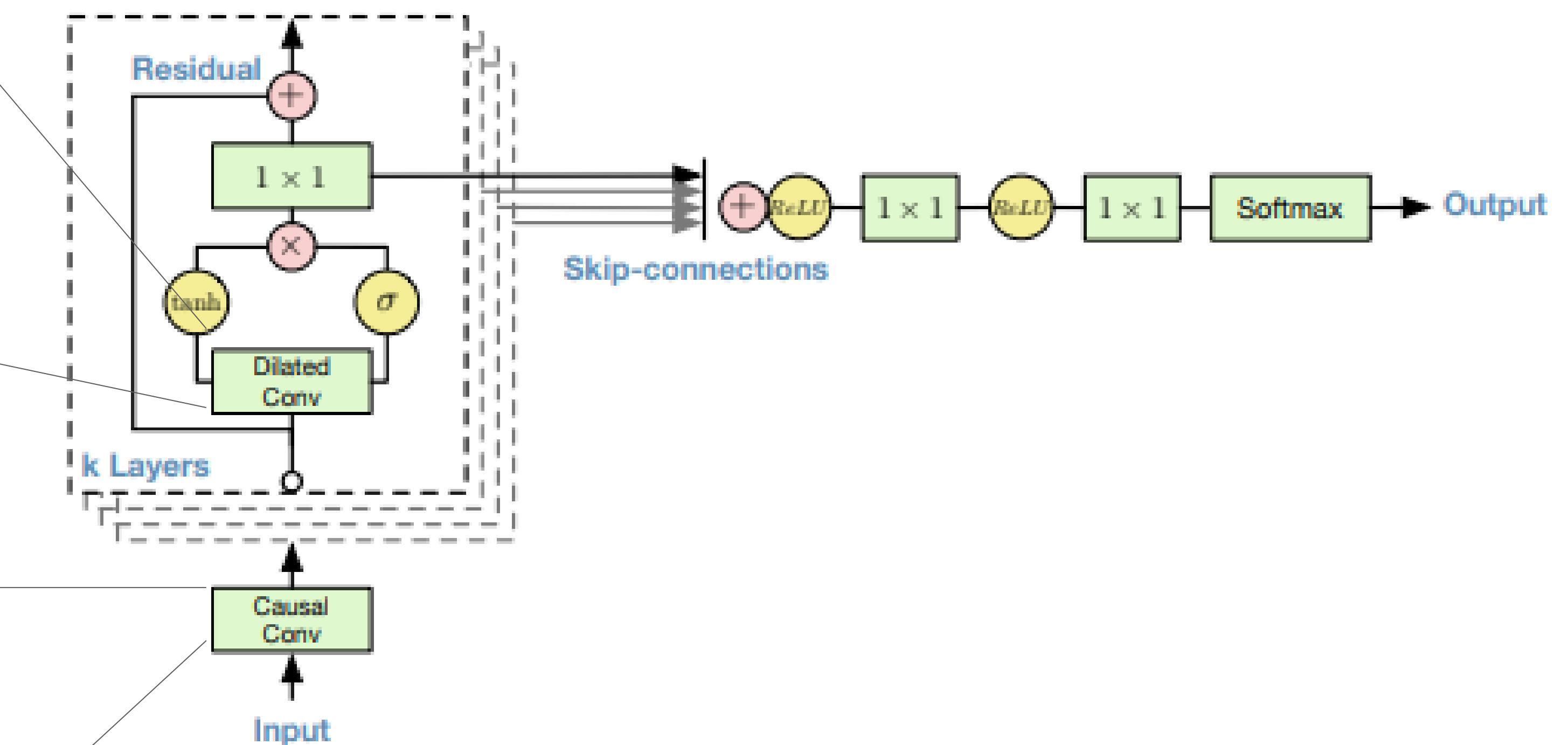
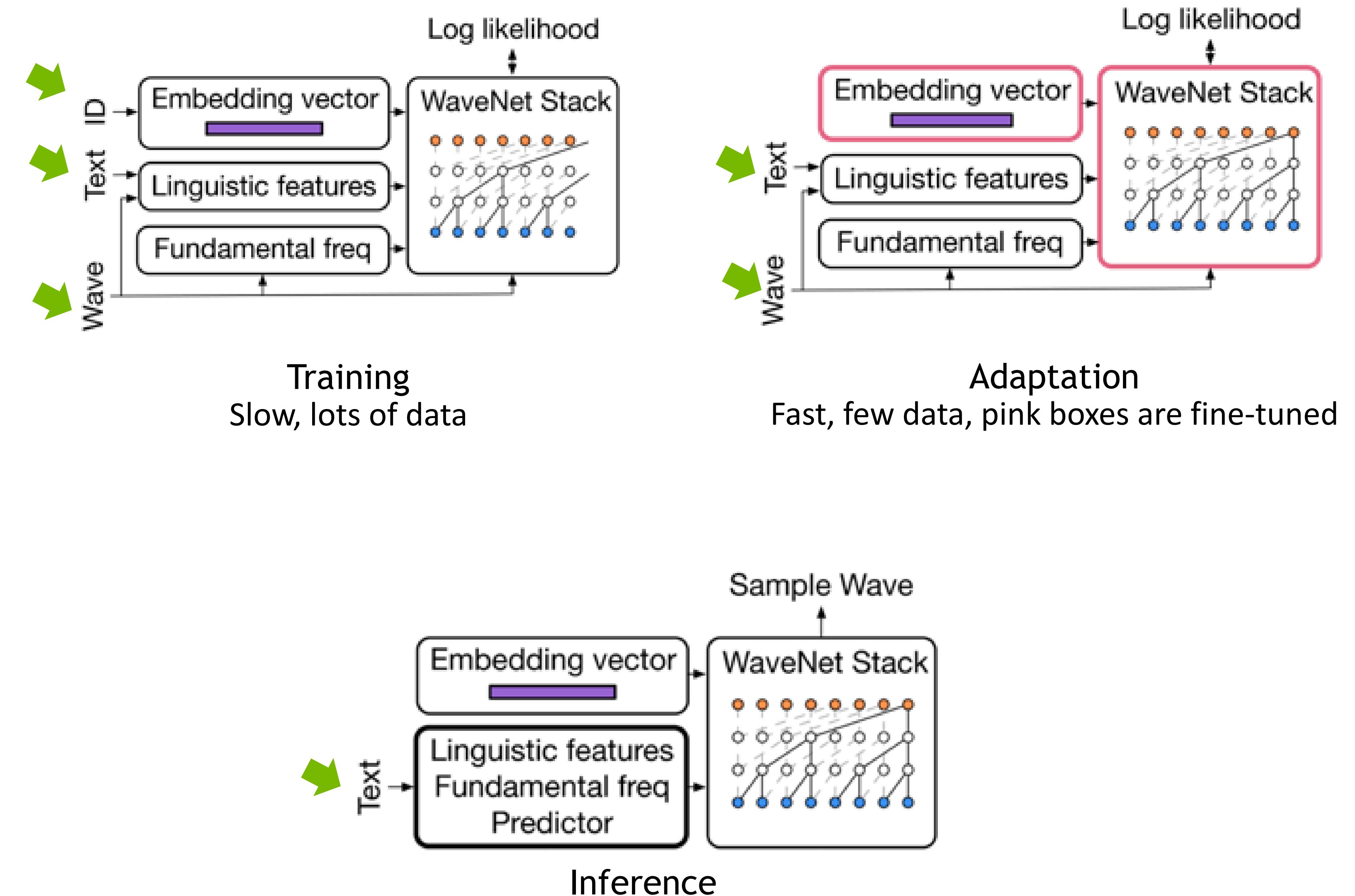


Figure 4: Overview of the residual block and the entire architecture.

End-to-End Synthesis

WaveNet - autoregressive



Speech samples	Subjective 5-scale MOS in naturalness	
	North American English	Mandarin Chinese
LSTM-RNN parametric	3.67 ± 0.098	3.79 ± 0.084
HMM-driven concatenative	3.86 ± 0.137	3.47 ± 0.108
WaveNet (L+F)	4.21 ± 0.081	4.08 ± 0.085
Natural (8-bit μ -law)	4.46 ± 0.067	4.25 ± 0.082
Natural (16-bit linear PCM)	4.55 ± 0.075	4.21 ± 0.071

Table 1: Subjective 5-scale mean opinion scores of speech samples from LSTM-RNN-based statistical parametric, HMM-driven unit selection concatenative, and proposed WaveNet-based speech synthesizers, 8-bit μ -law encoded natural speech, and 16-bit linear pulse-code modulation (PCM) natural speech. WaveNet improved the previous state of the art significantly, reducing the gap between natural speech and best previous model by more than 50%.

TTS Models

Recap

	Tasks	Approach	
DeepVoice 3	Spectrogram Generator	Convolutional	Autoregressive
Tacotron /Tacotron 2	Spectrogram Generator	Convolutional + LSTM	Autoregressive
FastSpeech	Spectrogram Generator	Transformer + Convolutional Condition to Duration and Pitch	Non Autoregressive
FastPitch	Spectrogram Generator	Transformer + Convolutional Condition to Duration and Pitch	Non Autoregressive
TalkNet	Spectrogram Generator	Convolutional Condition to Duration and Pitch	Non Autoregressive
Glow TTS	Spectrogram Generator	Transformer + Convolutional Condition to Duration and Pitch	Non Autoregressive
RadTTS	Spectrogram Generator	Generative modelling of durations to add diversity	Non Autoregressive
Flowtron	Spectrogram Generator	Modified Tacotron2 Condition to Speaker Embeddings and Speech Style	Autoregressive
WaveGlow	Vocoder	Affine Coupling Layer + Convolutional	Non Autoregressive
SqueezeWave	Vocoder	Affine Coupling Layer + Convolutional	Non Autoregressive
HifiGAN	Vocoder	Convolutional + MRF Adversarial Training: 1 Generator and 2 discriminators	Non Autoregressive
Wavenet	End-to-End synthesis	Dilated Convolutions Linguistic conditioning, F0 and duration	Autoregressive



TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

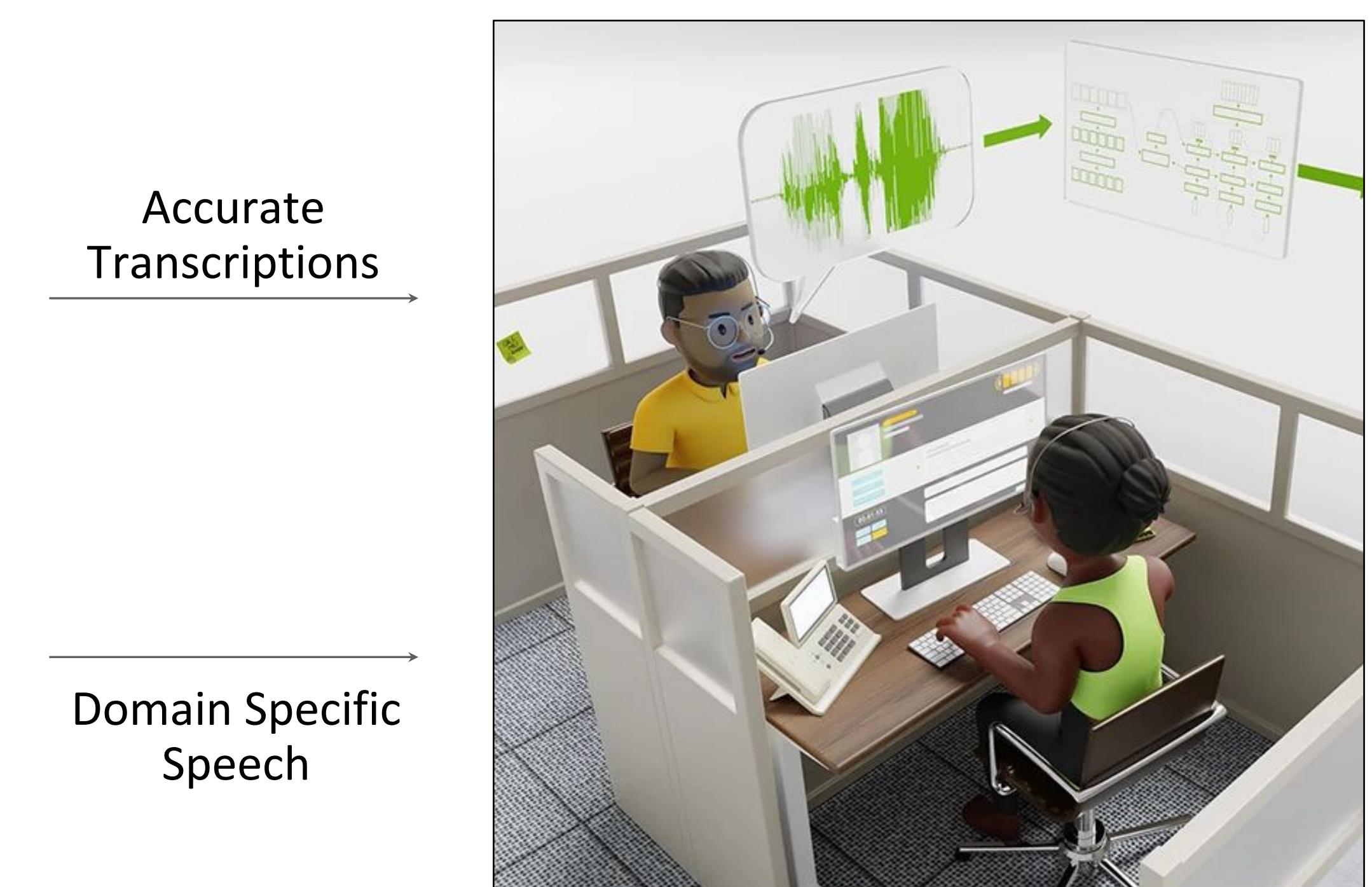
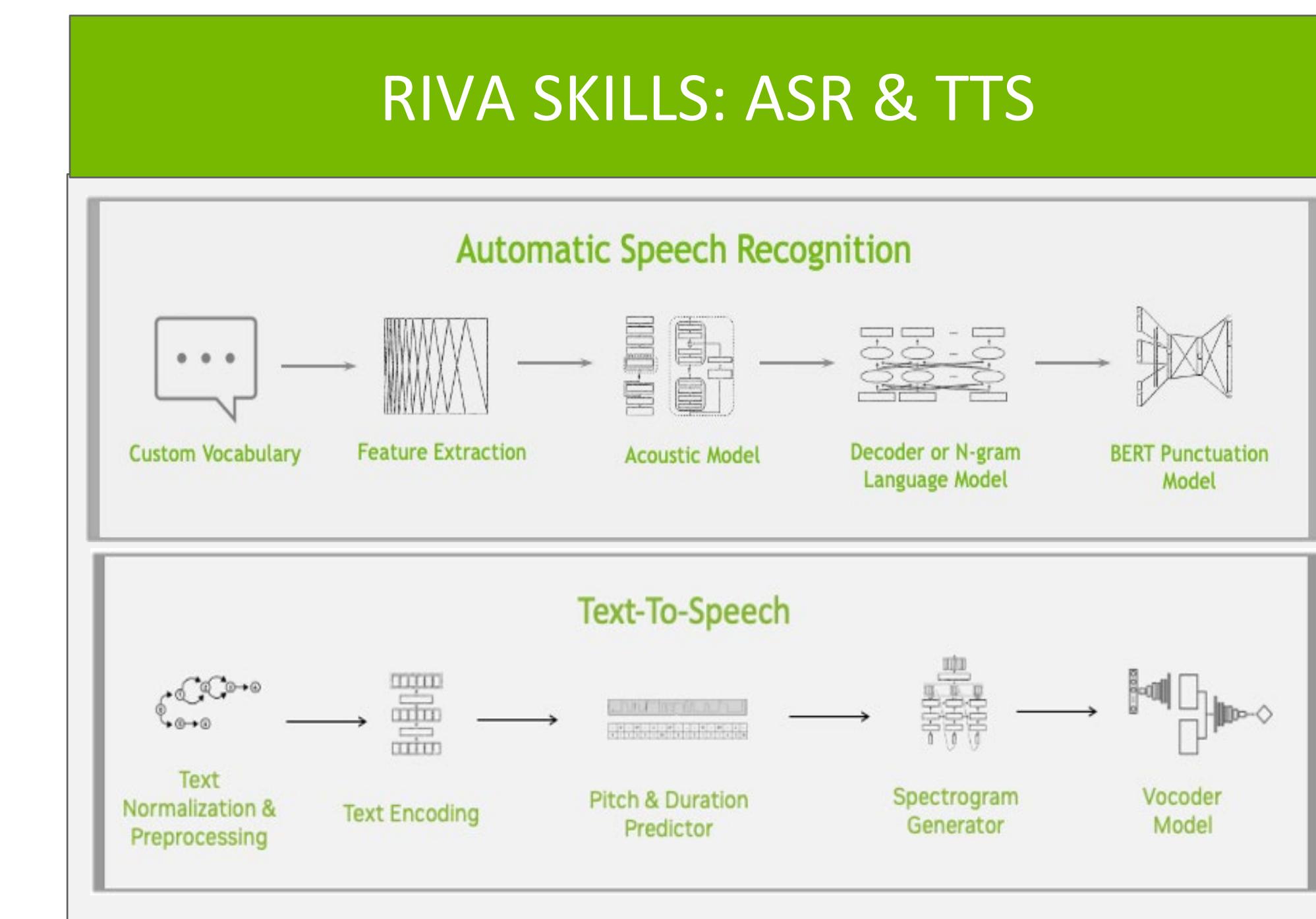
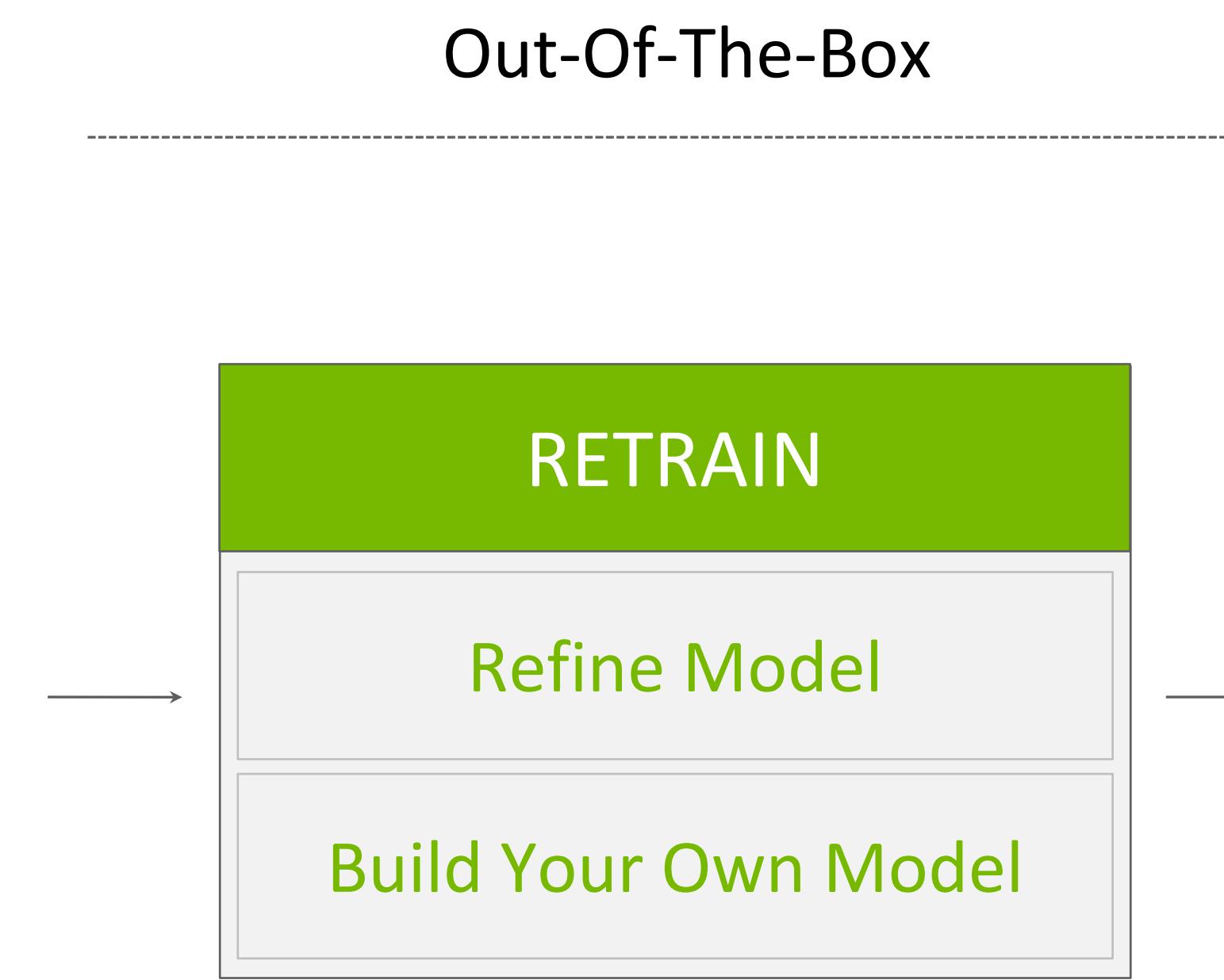
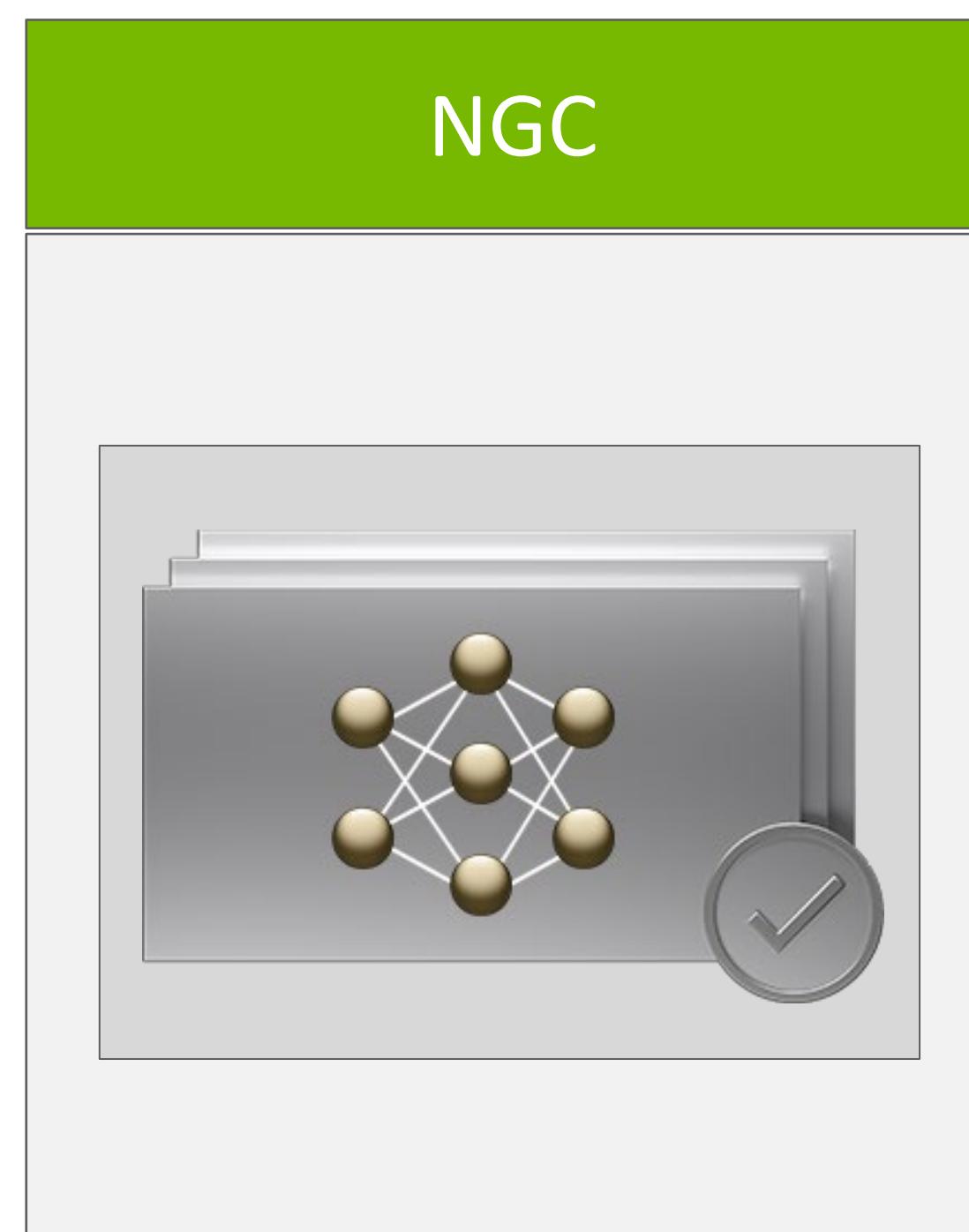
- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Part 2: Lab

- ASR, TTS and NLP Customization

NVIDIA Riva

Fully customizable, GPU-Accelerated SDK for real-time Speech AI



- Fully Customizable for the Best Possible Accuracy
- GPU-Accelerated, Real-Time
- Scale to Hundreds of Thousands of Users
- On-Prem, in Any Cloud, at the Edge, or Embedded

SOTA – State-Of-The-Art

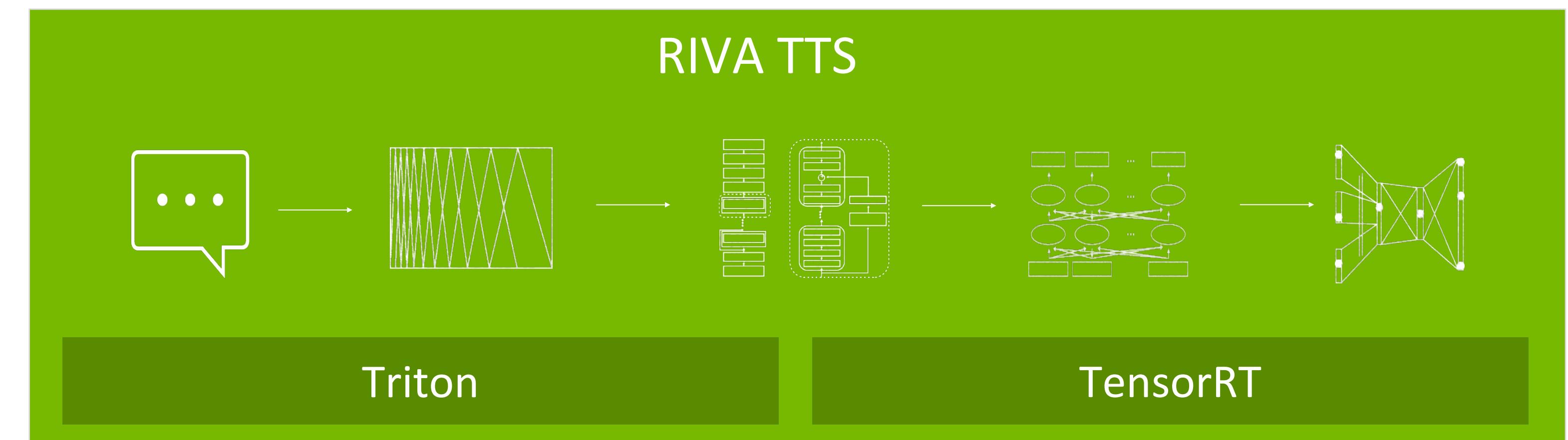
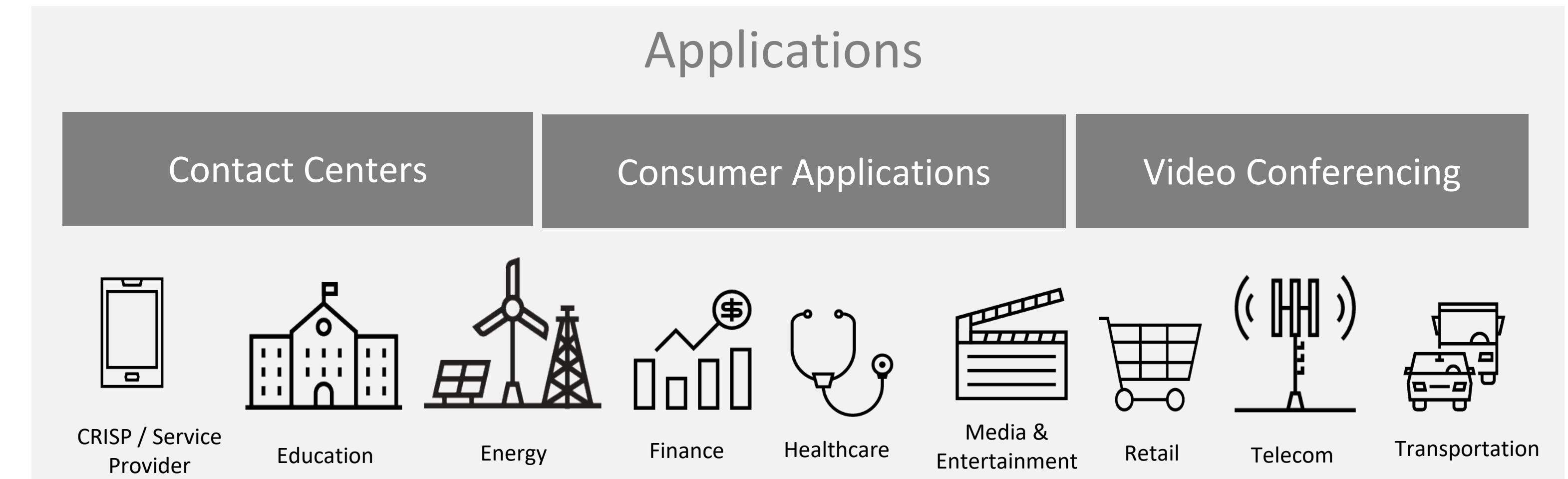
| ASR – Automatic Speech Recognition

| TTS – Text-To-Speech

NVIDIA Riva Text-To-Speech

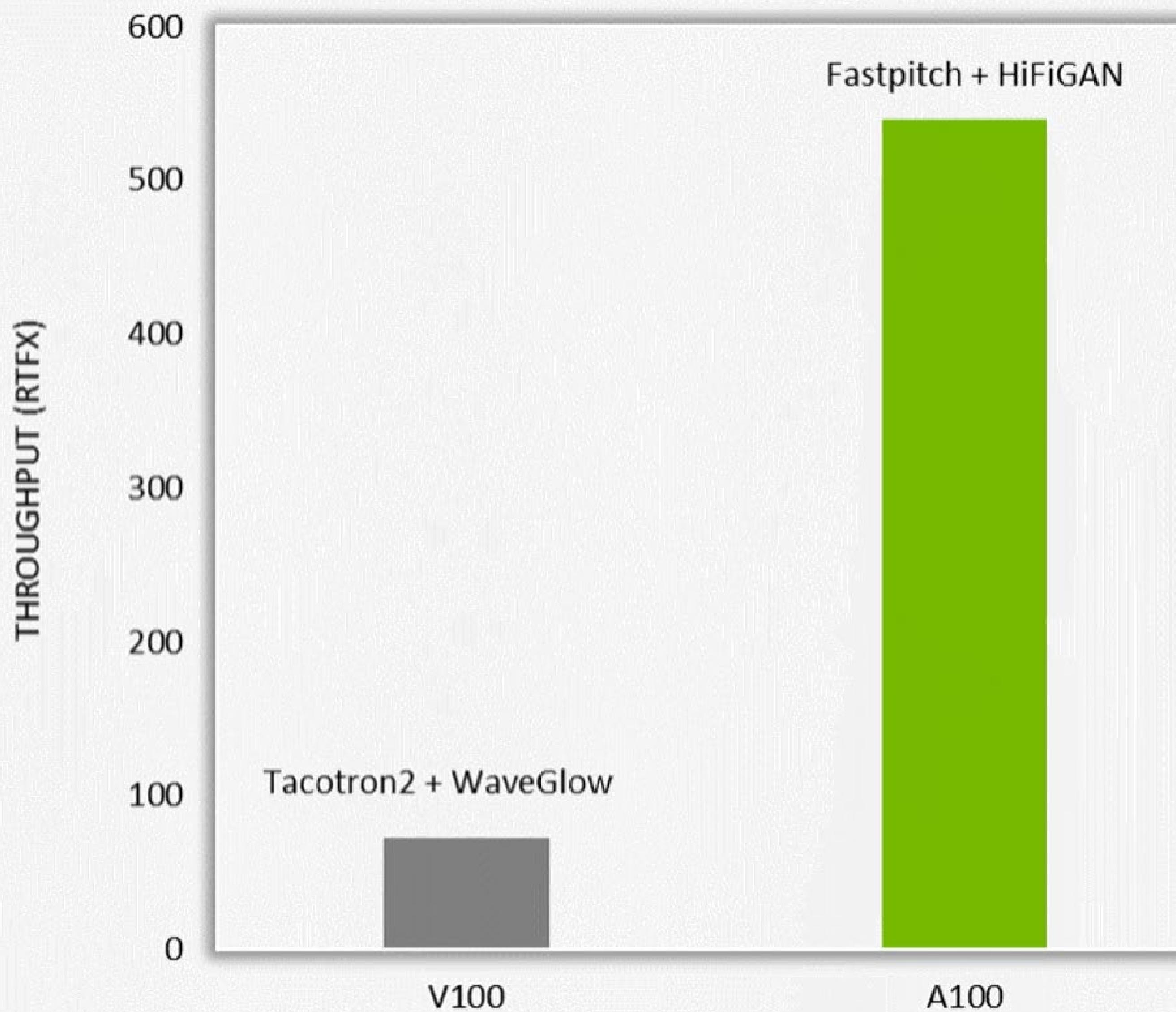
World-Class TTS with SOTA Deep Learning

- **SOTA OOTB expressive voice models**
- Female & male **OOTB professional voices** for English
- **Easy voice control** at inference time with SSML API for:
 - Pronunciation
 - Pitch
 - Volume
 - Pause
- **12X higher inference performance** versus existing technologies.



NVIDIA Riva Delivers World-Class TTS

12X Performance with SOTA TTS Models



HIGH-QUALITY TTS VOICES



US English Female



US English Male

TTS – Text-To-Speech

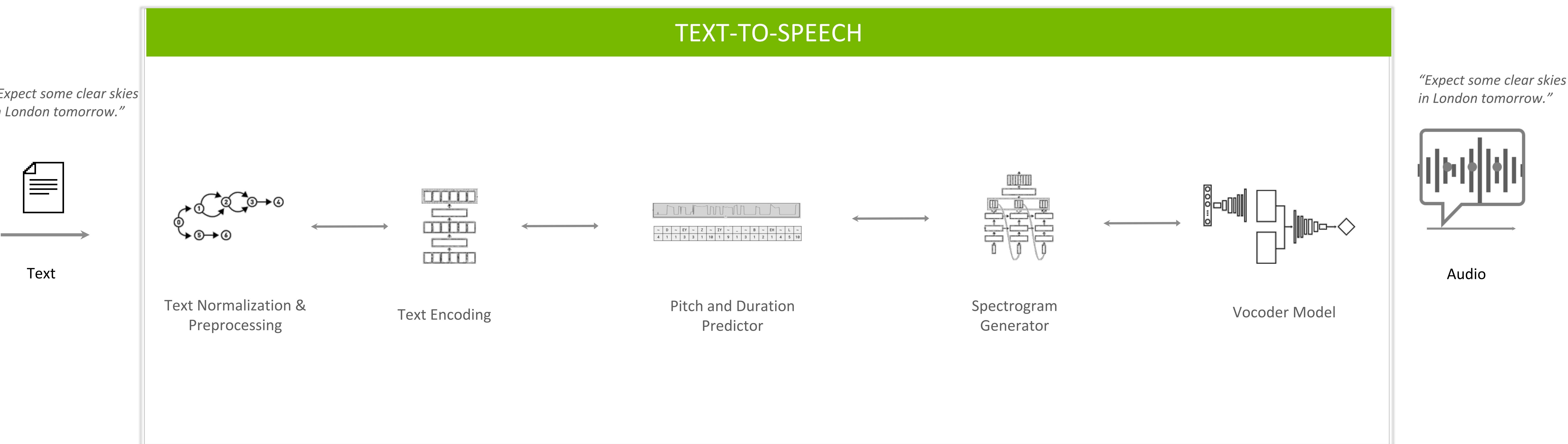
| SOTA – State-Of-The-Art

| RTFX – Inverse of Real Time Factor (RTF) = $\frac{\text{compute time [sec]}}{\text{audio inferred [sec]}}$



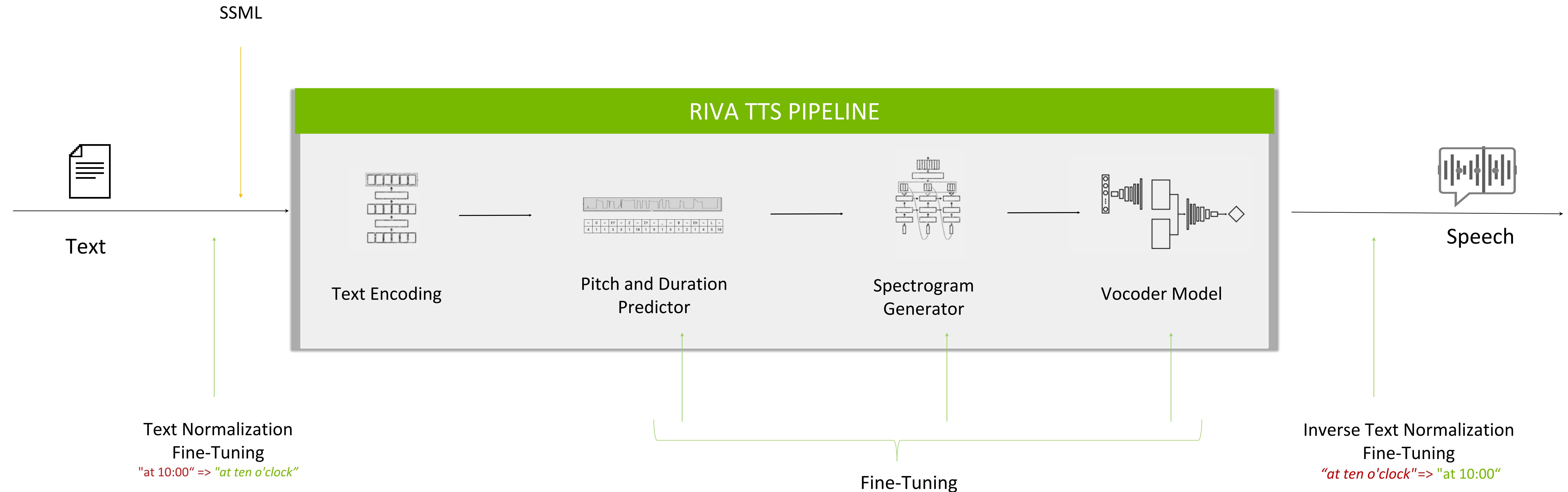
Achieving Human-Like Voice and Intonation for Real-World TTS Apps Is Challenging

Sounding natural, emotion expressivity, compute intensive pipeline robustness



Customizations Across Riva TTS Pipeline

Controllability, expressiveness, high-performance



Customization during:

- Training (server-side) – Need to rebuild/restart server
- Inference (client-side) – No need to rebuild/start Riva server

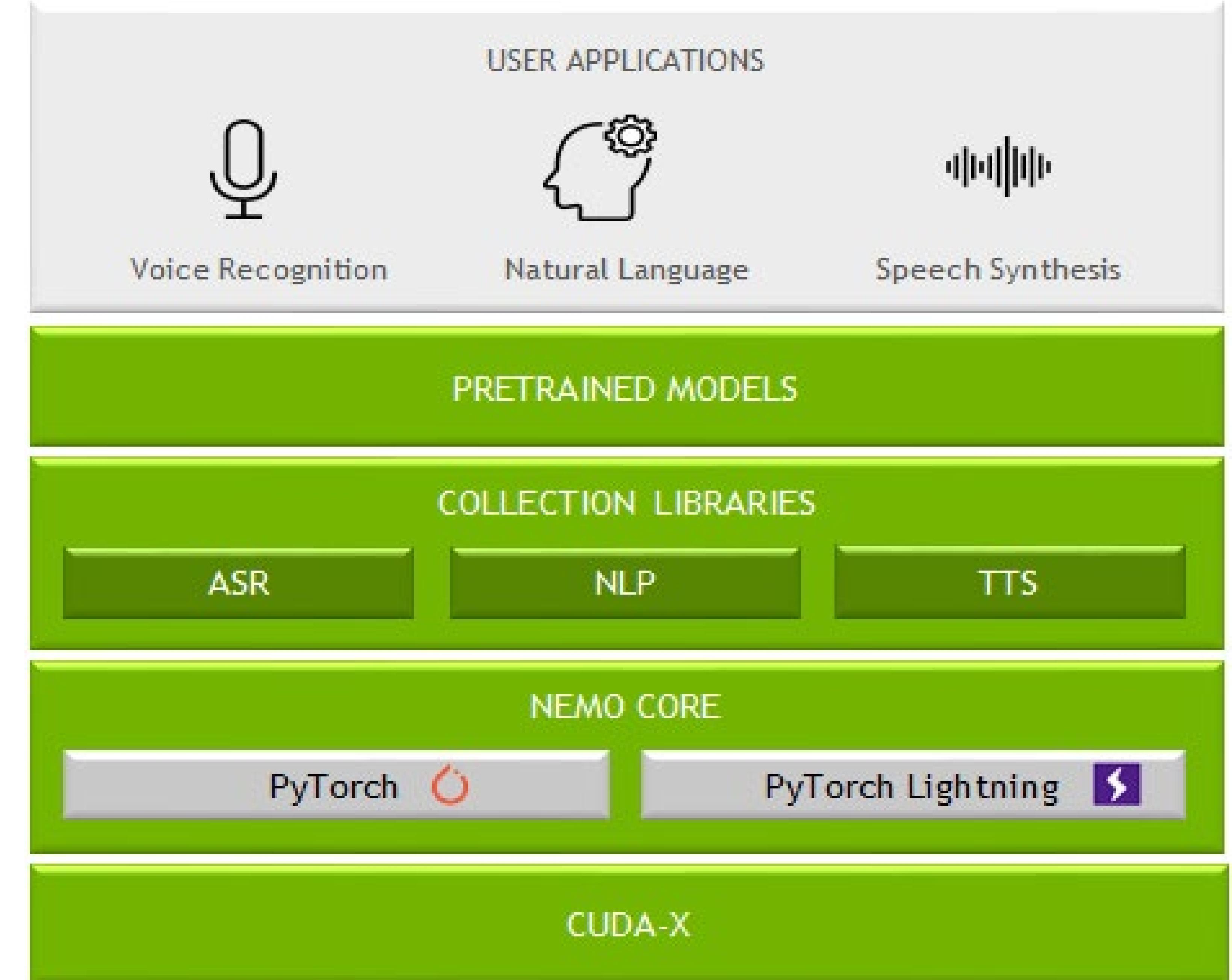
TTS – Text-To-Speech | SSML – Speech Synthesis Markup Language



What is NeMo?

Toolkit to build state-of-the-art conversational AI models

- DL-Based Speech & Language Understanding Models
- Include Semantic Checking for Correct-by-Construction Architectures
- Open-Sourced
- Integrated with PyTorch & PyTorch Lightning
- Easy-to-Use APIs
- Optimized Training Performance
- 100+ Pre-Trained GPU-Optimized Checkpoints
- Scale to 1000s of NVIDIA GPUs



<https://ngc.nvidia.com/catalog/containers/nvidia:nemo>

Deploy to Production

Generate high-performance inference

- Quickly export NeMo models to Riva
- Support for speech and language models across multiple languages
- Step-by-step deployment instructions in documentation



Exporting NeMo Models to Riva



TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Part 2: Lab

- ASR, TTS and NLP Customization

The Goal of the Lab

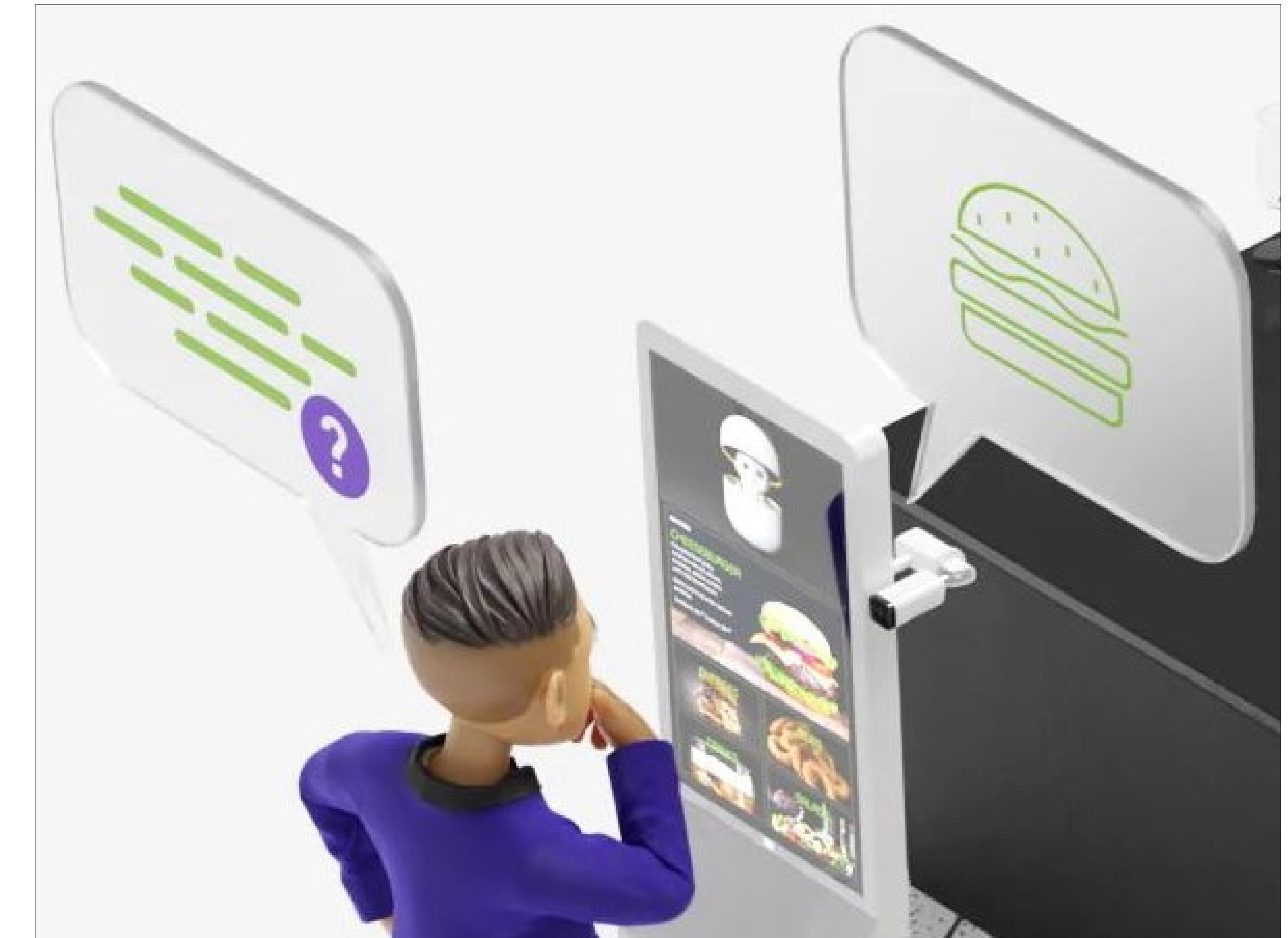
Beyond OOTB models

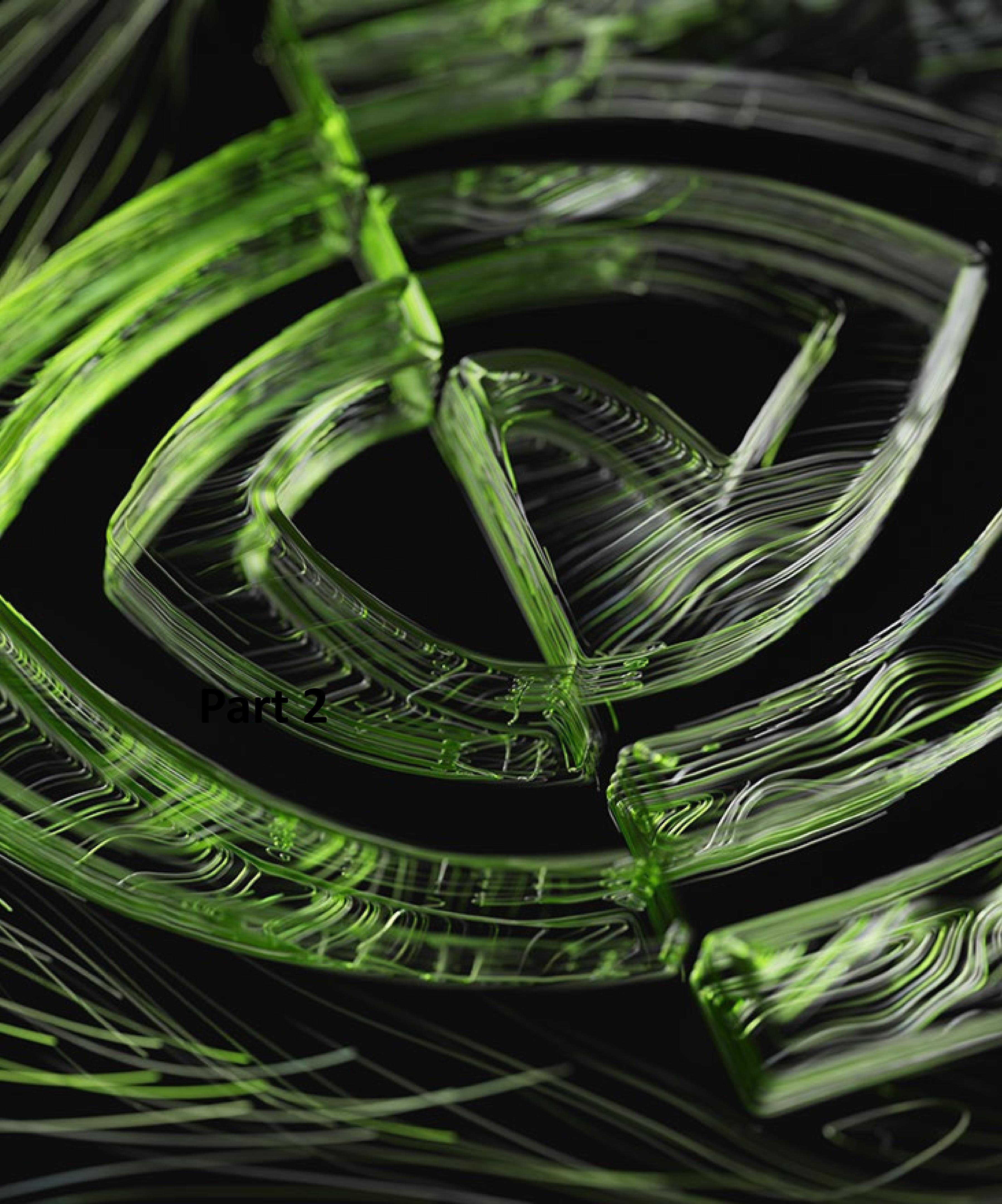
1. Investigate AI models for Speech Synthesis

- Normalize written text to speech-appropriate text and convert graphemes to phonemes
- Use the Fastpitch model to create a spectrograph
- Convert spectrograms to audio with the HifiGan vocoder model and Customize speech output

2. TTS deployment with Riva

- Launch Riva TTS service
- Request the TTS service using a Python client API
- Customize pronunciations





TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Part 2: Lab

- ASR, TTS and NLP Customization

!

Part 1: Lab



TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

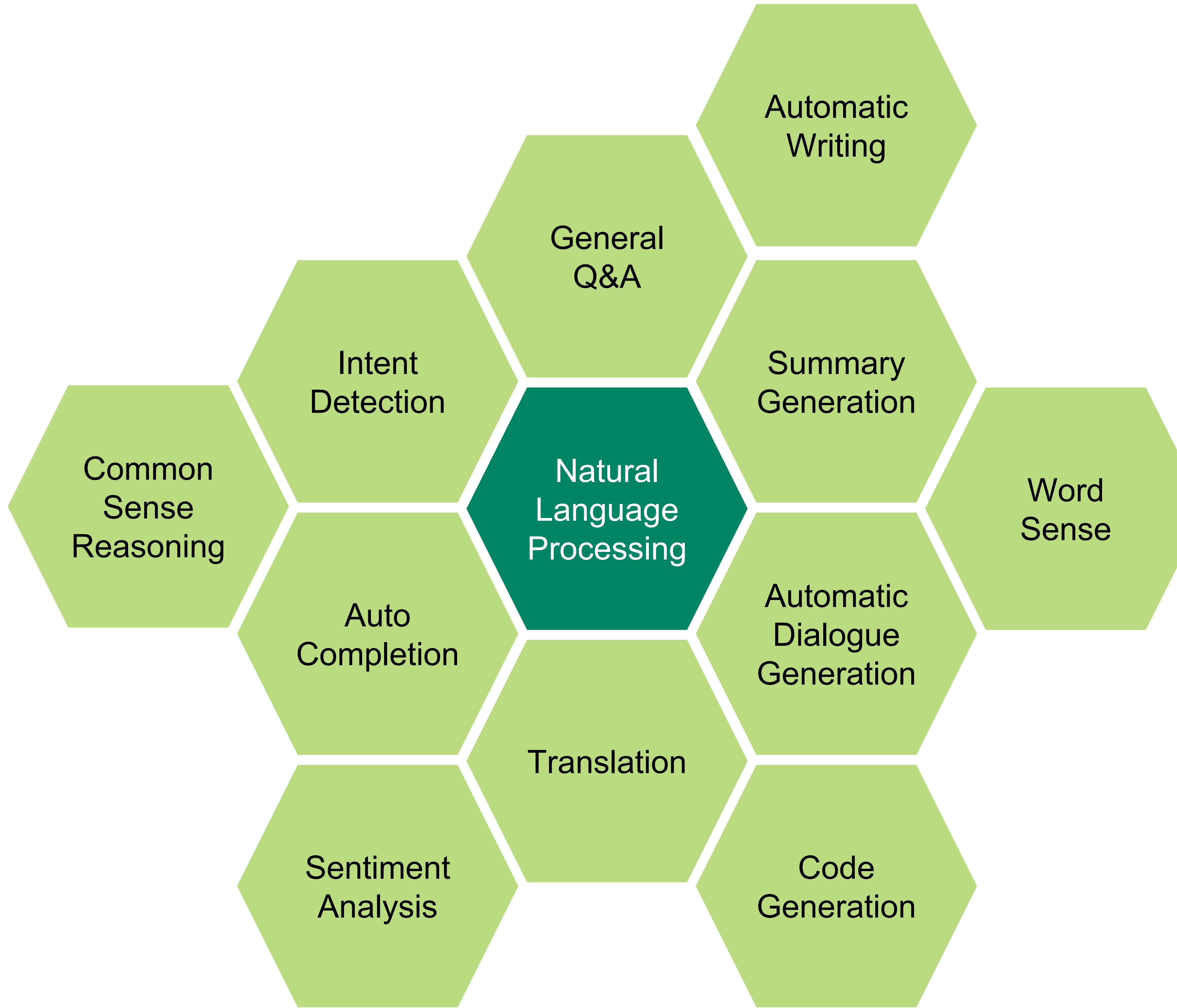
- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Part 2: Lab

- ASR, TTS and NLP Customization

!

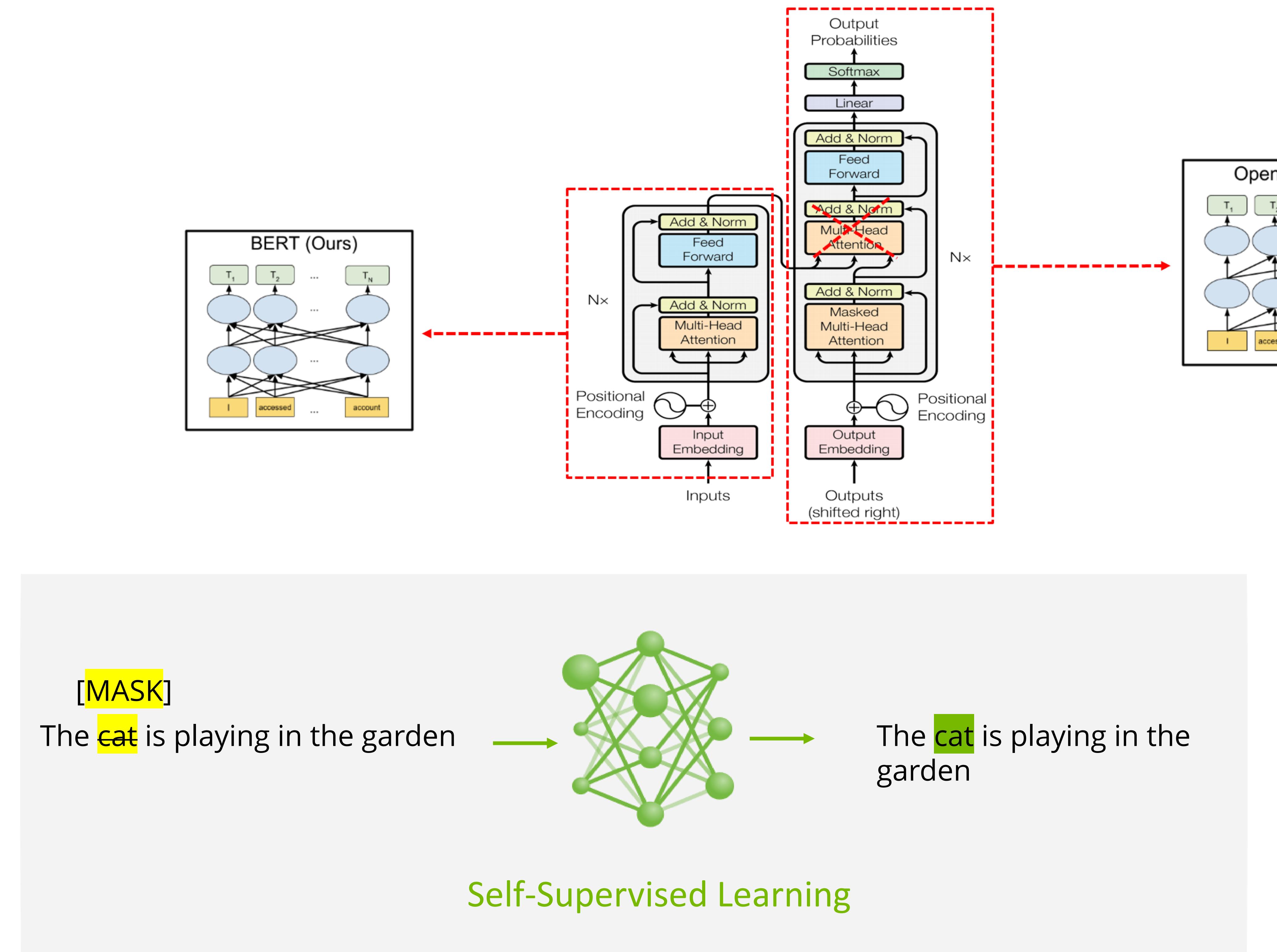
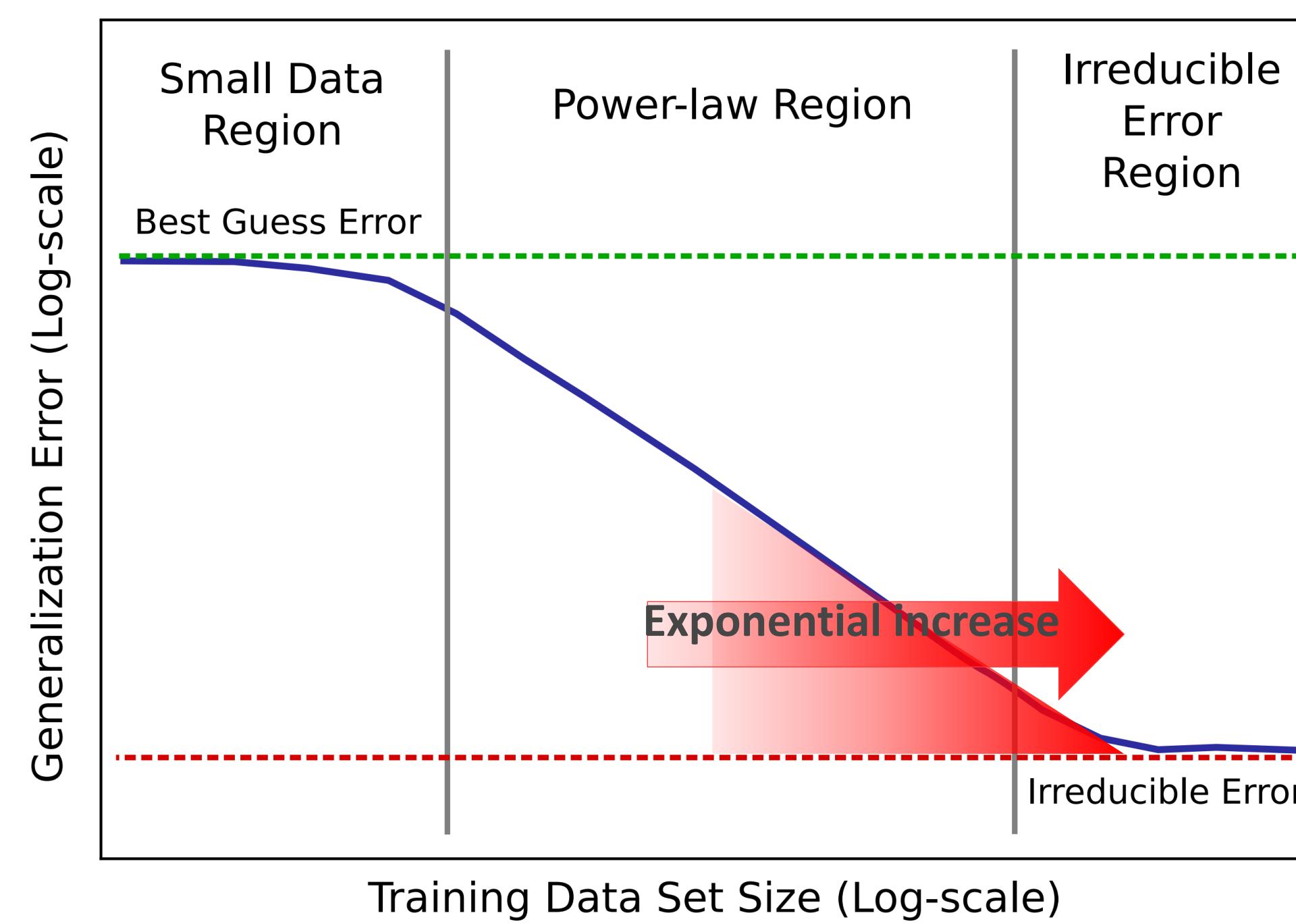
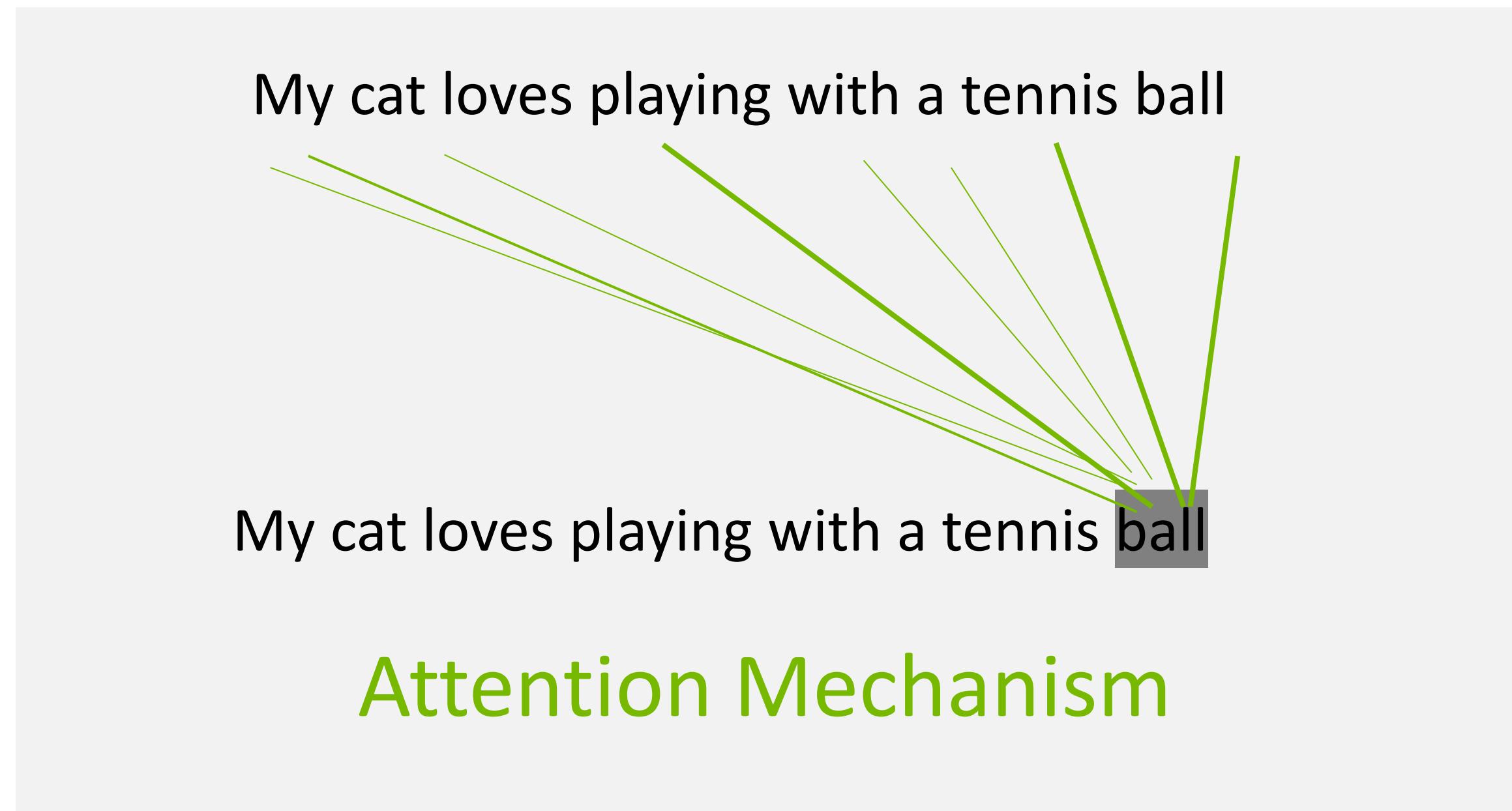
Setting Expectations: Focus on Model Related to Conversational AI



And many more....

Transformer Language Models

Reasons for success



General Purpose Language Model

Used for feature extraction

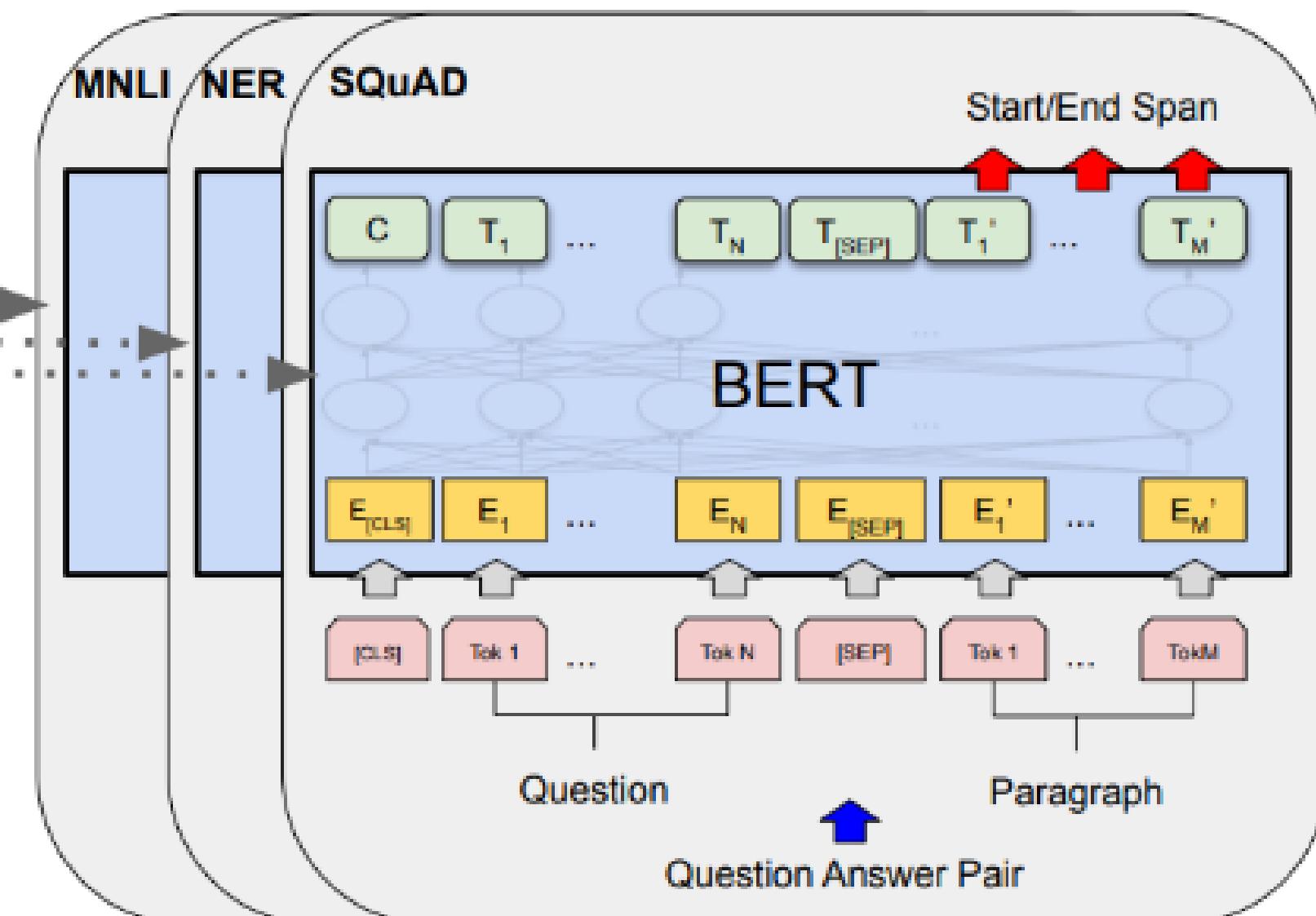
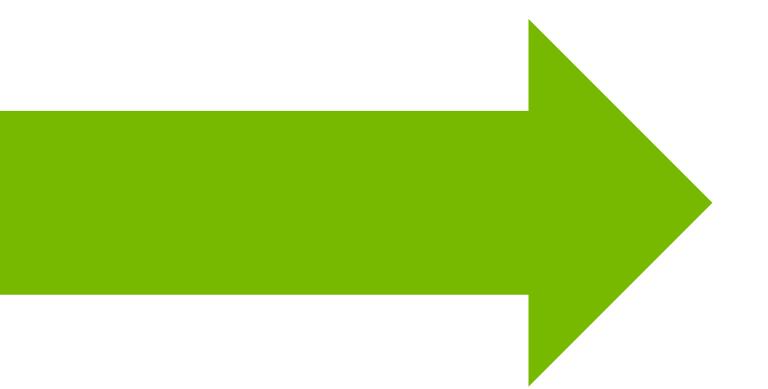
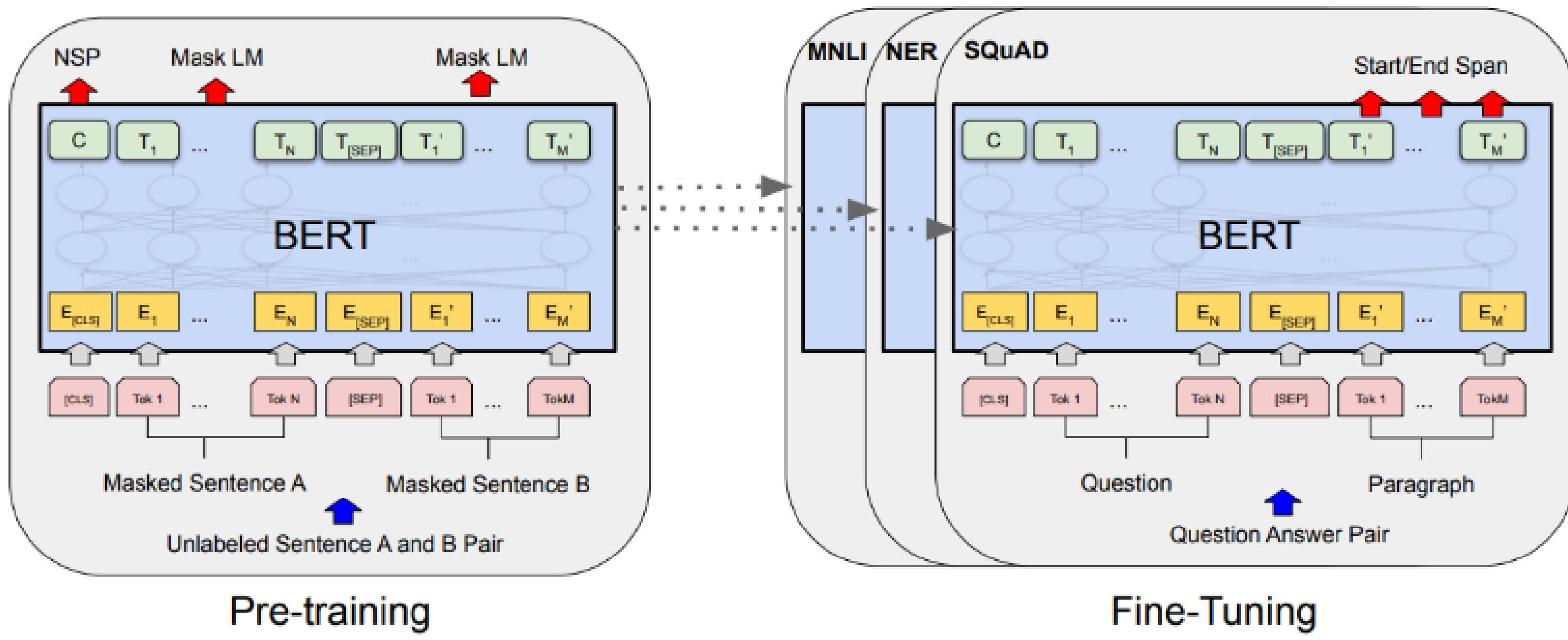
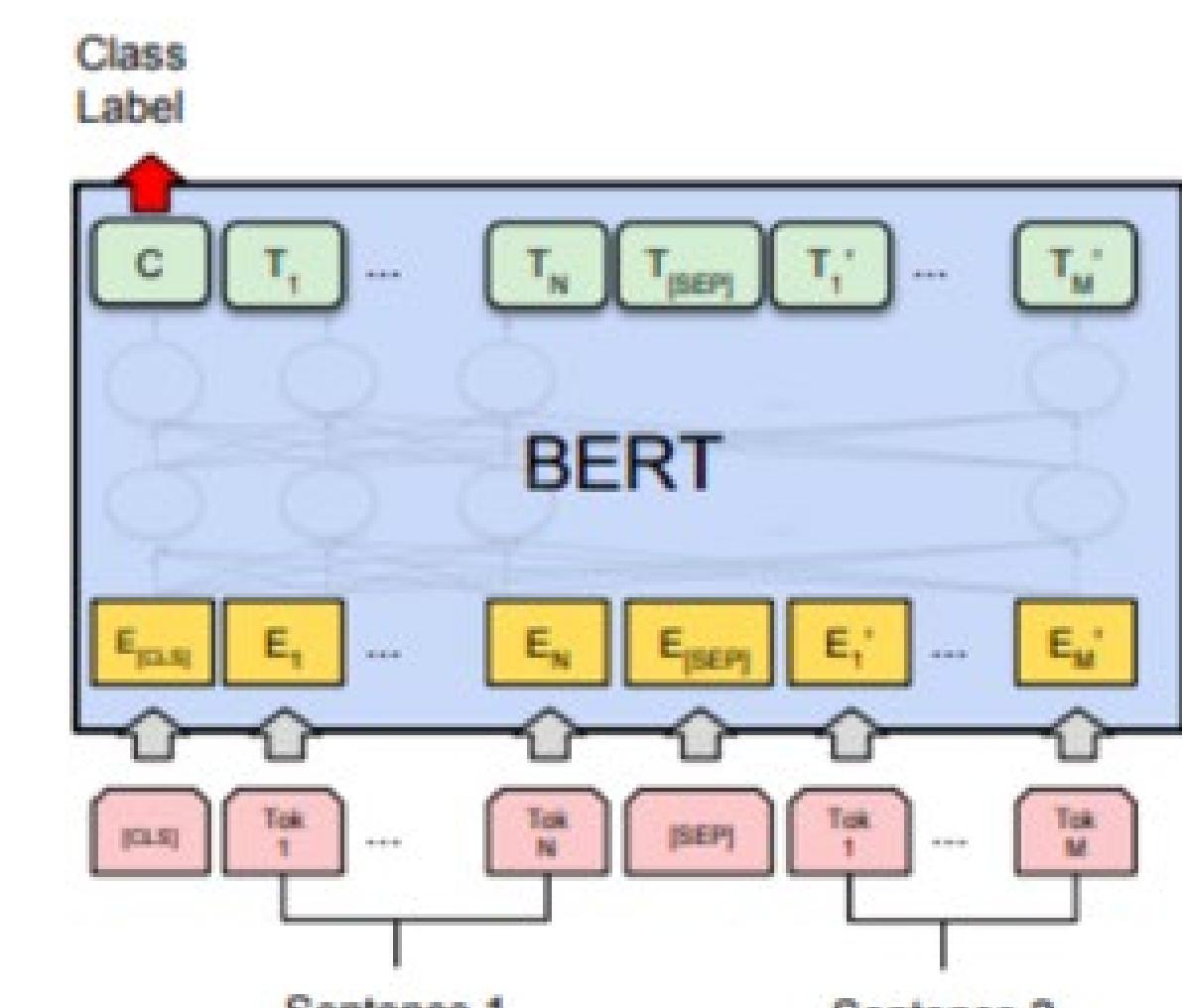
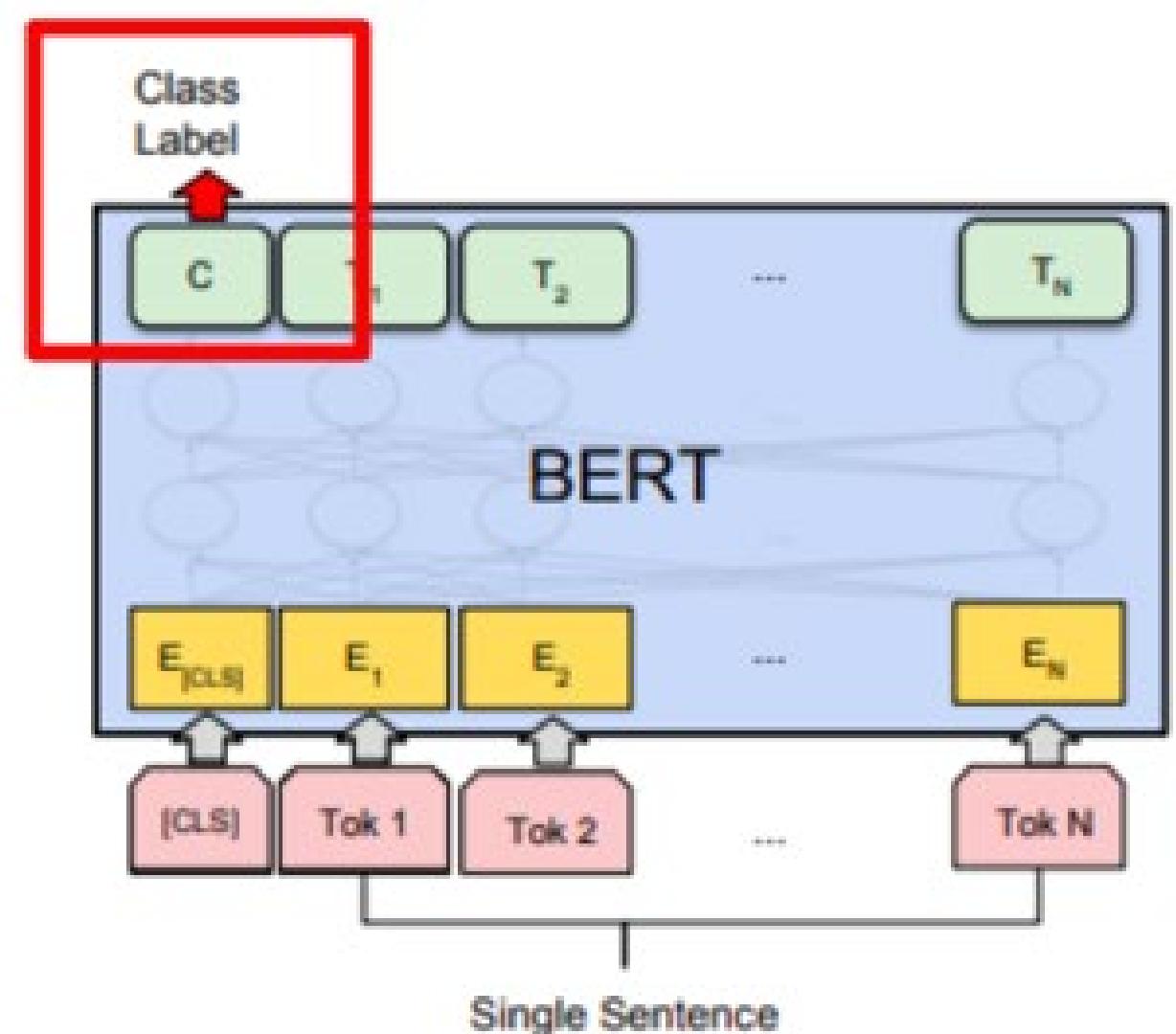


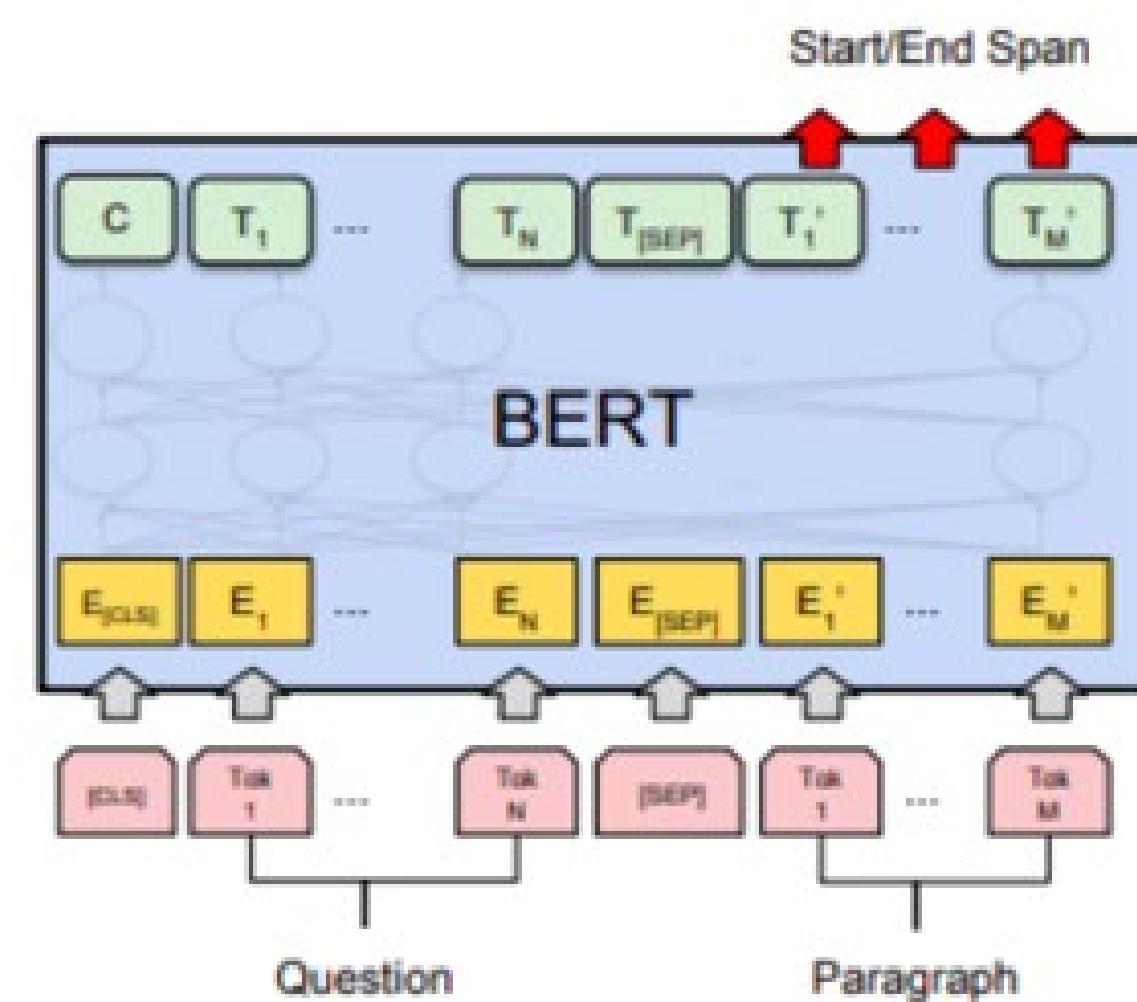
Figure 1: Overall pre-training and fine-tuning procedures for BERT. Apart from output layers, the same architectures are used in both pre-training and fine-tuning. The same pre-trained model parameters are used to initialize models for different down-stream tasks. During fine-tuning, all parameters are fine-tuned. [CLS] is a special symbol added in front of every input example, and [SEP] is a special separator token (e.g. separating questions/answers).



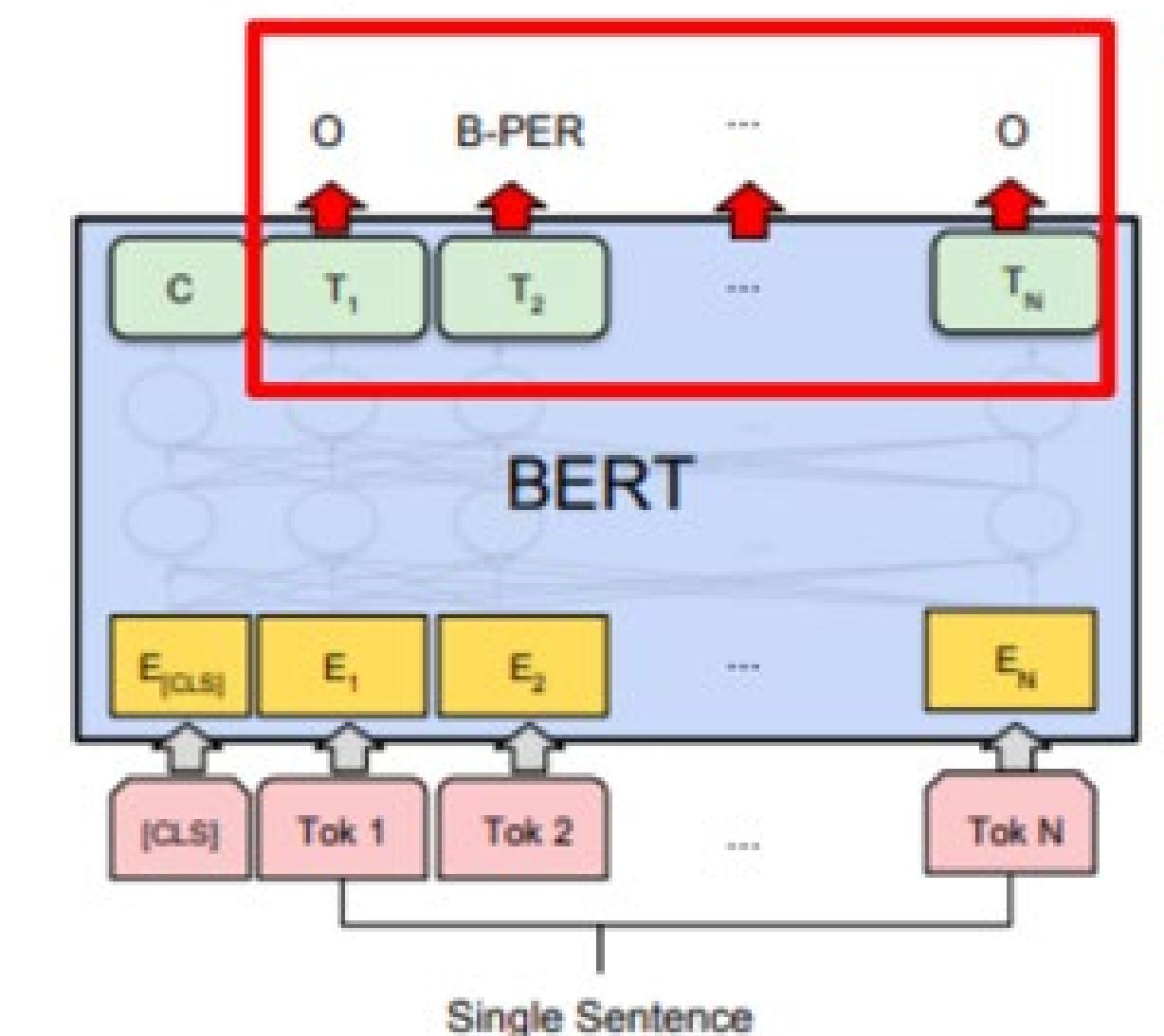
(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG



(b) Single Sentence Classification Tasks:
SST-2, CoLA



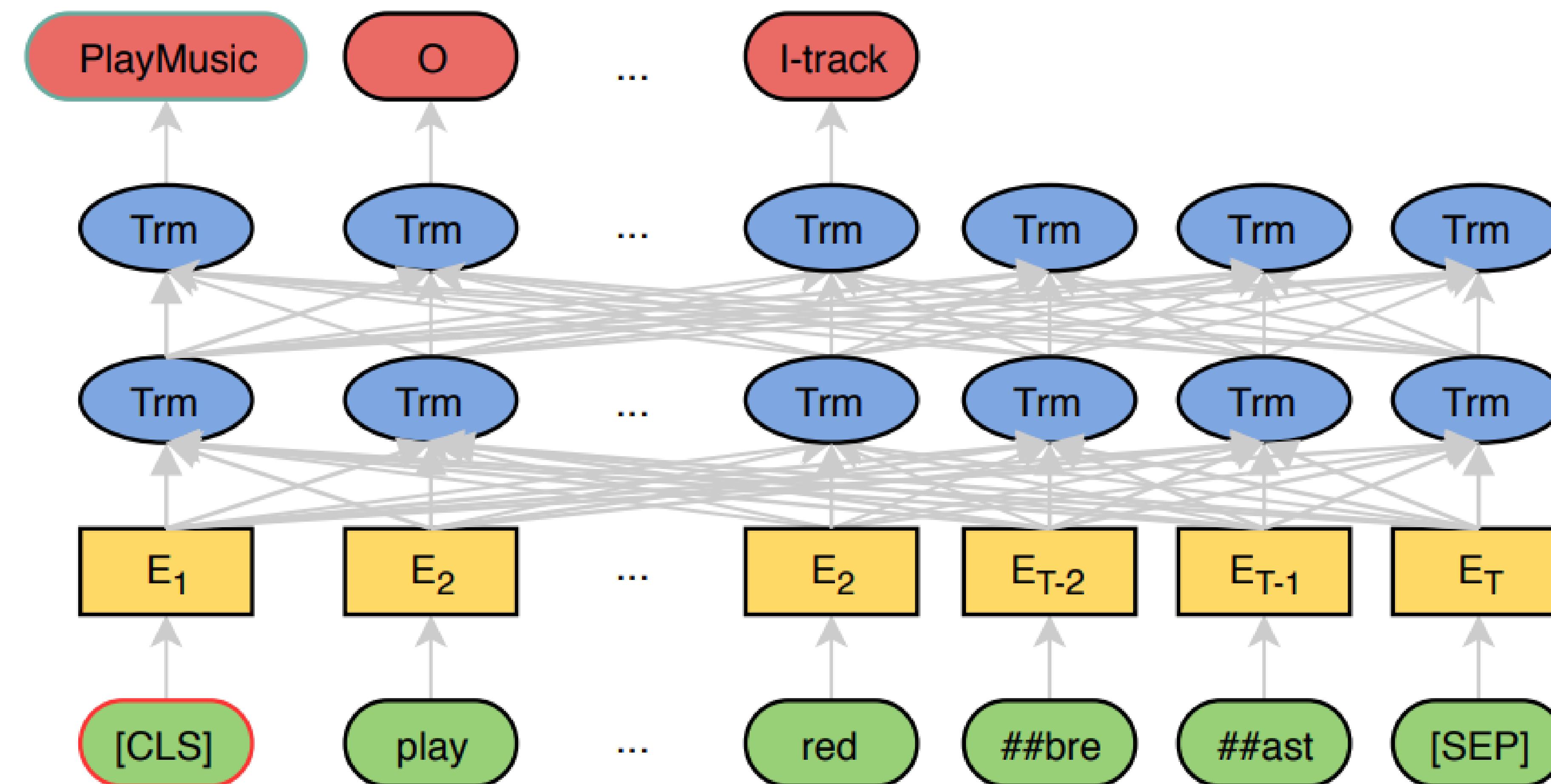
(c) Question Answering Tasks:
SQuAD v1.1



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

Reusing Language Models

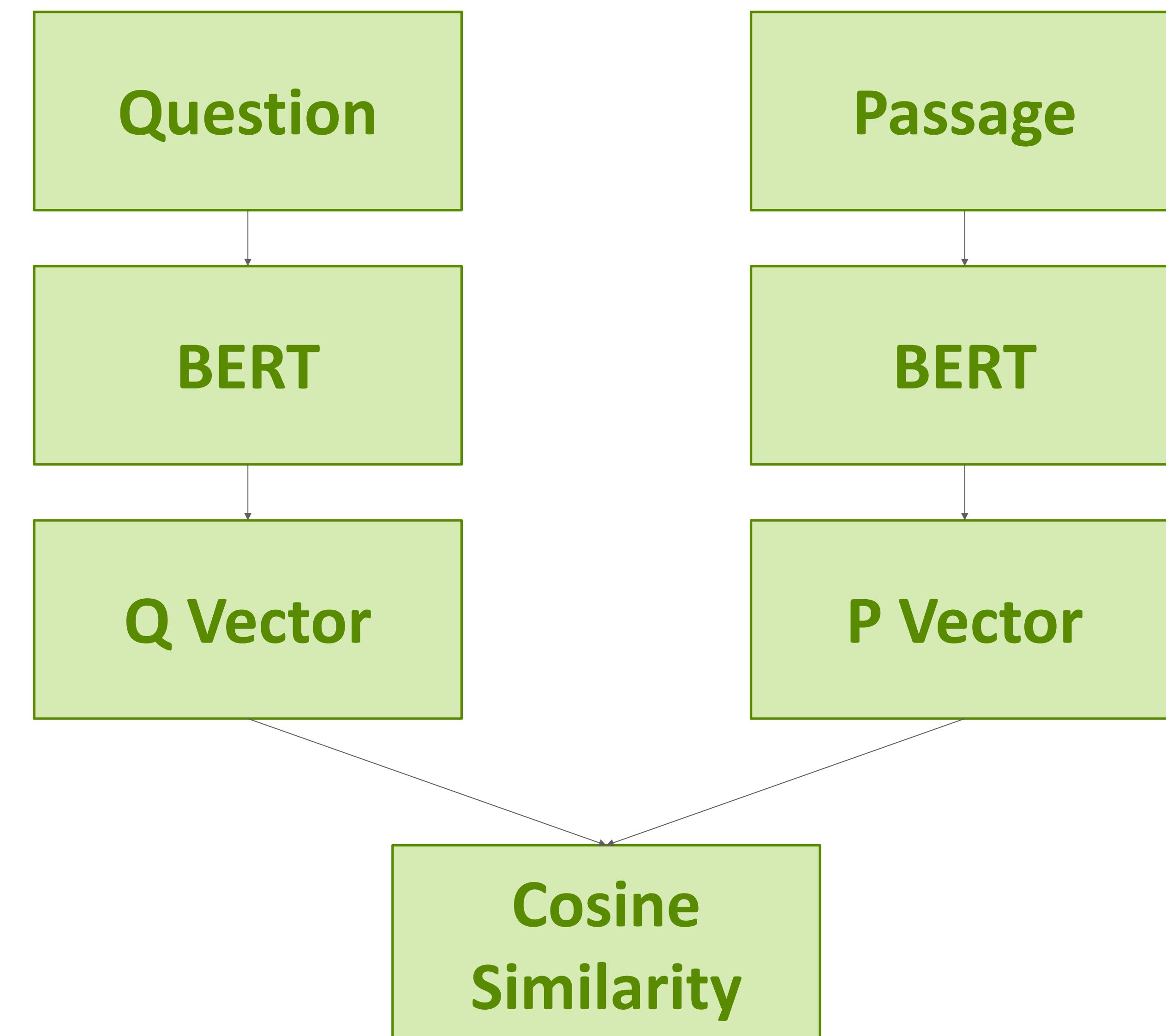
Joint Intent classification and slot filling

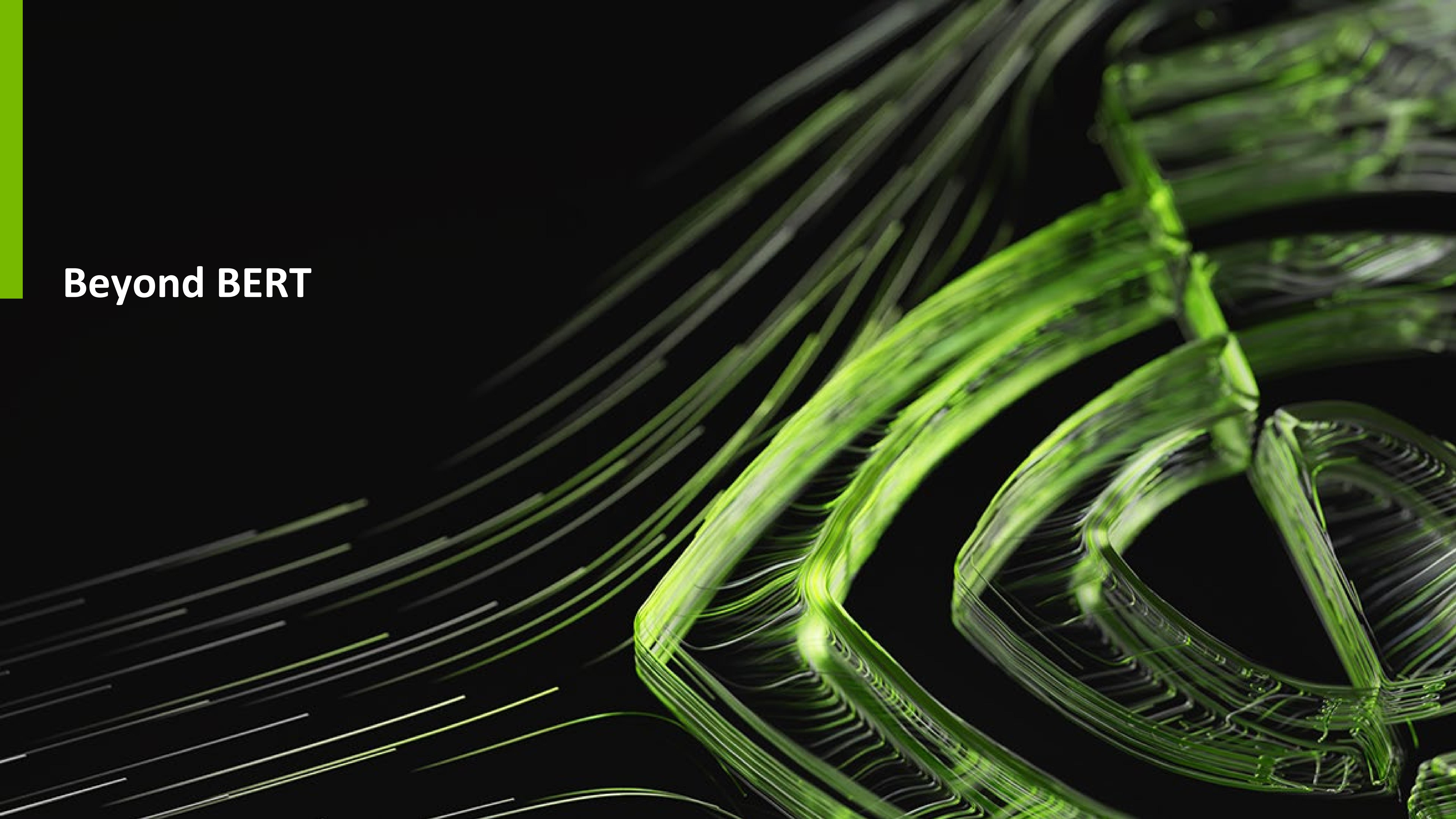


The input query is “*play the song little robin redbreast*”

Information Retrieval

Dense passage retrieval

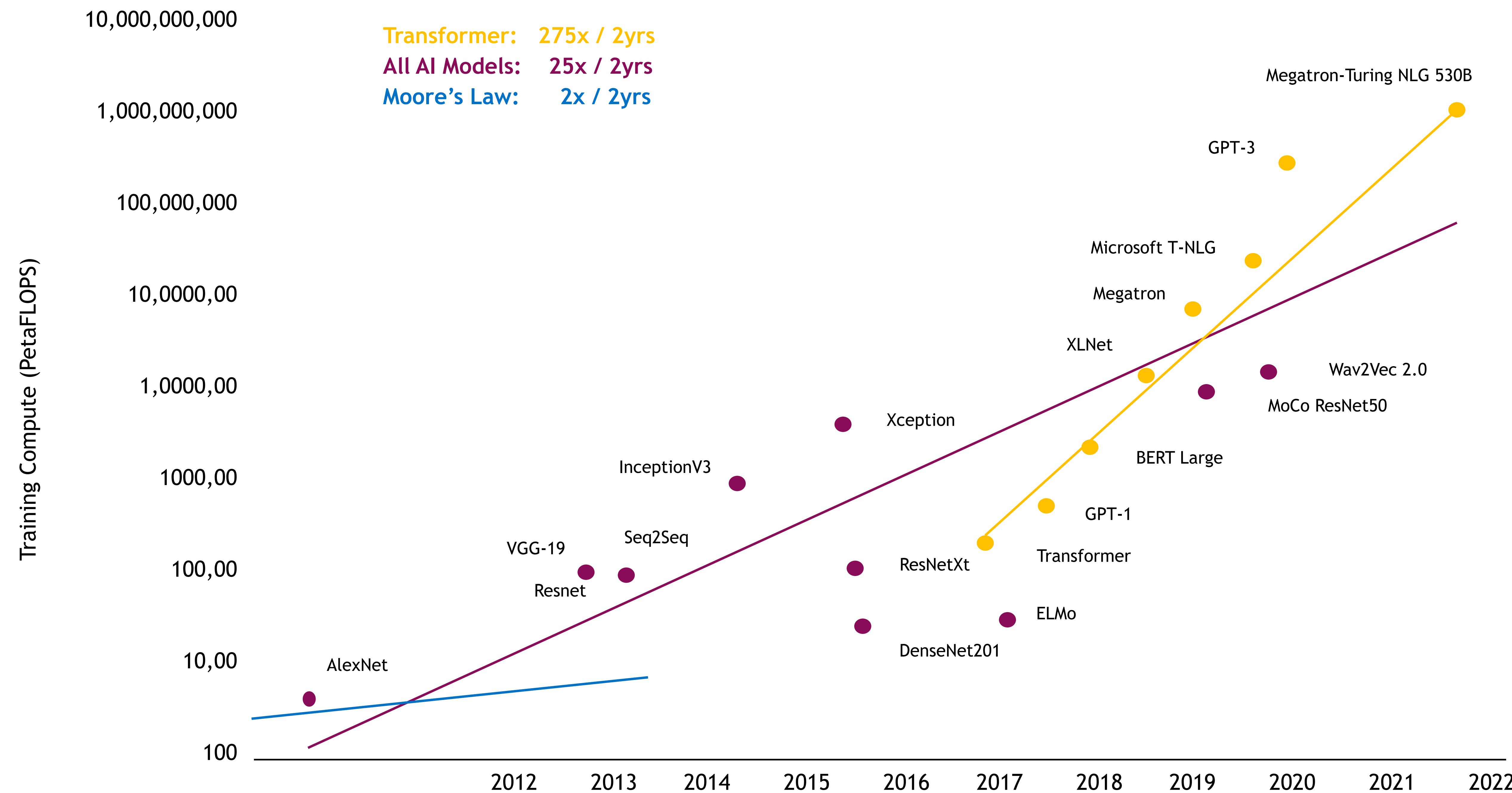




Beyond BERT

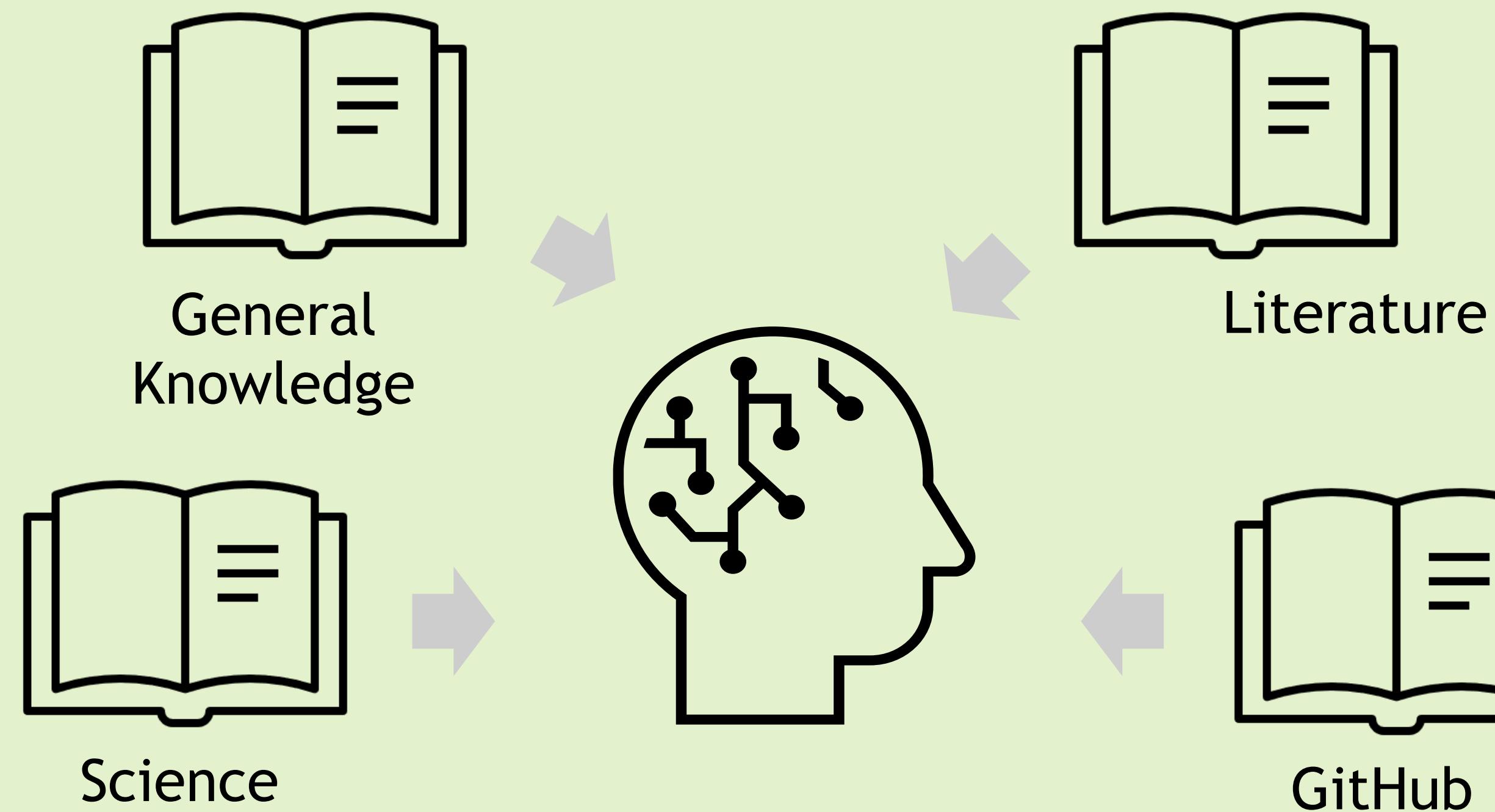
Very Large Models Are the Presence of AI

Beyond NLP



Fundamental Change in the NLP Paradigm

Step 1: Train a large scale model



Step 2. Ask questions

‘Q: Would you say this movie
review is positive or negative?
“I loved that movie”’

→ → ‘A: Negative’

Downstream Tasks

Zero/few shot learners

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

- 1 Translate English to French: ← task description
- 2 cheese => ← prompt

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

- 1 Translate English to French: ← task description
- 2 sea otter => loutre de mer ← example
- 3 cheese => ← prompt

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

- 1 Translate English to French: ← task description
- 2 sea otter => loutre de mer ← examples
- 3 peppermint => menthe poivrée
- 4 plush girafe => girafe peluche
- 5 cheese => ← prompt

Downstream Tasks

Prompt Engineering

Type	Task	Input ([X])	Template	Answer([Y])
	Sentiment	I love this movie.	[X] The movie is [Y]	great fantastic ...
Text CLS	Topics	He prompted the LM.	[X] The text is about [Y]	sports science ...
	Intention	What is taxi fare to Denver?	[X] The question is about [Y]	quantity city ...
Text-span CLS	Aspect Sentiment	Poor service but good food.	[X] What about service? [Y]	Bad Terrible ...
Text-pair CLS	NLI	[X1]: An old man with ... [X2]: A man walks ...	Hypothesis: [X1], Premise: [X2], Answer: [Y]	Contradiction Entailment ...
Tagging	NER	[X1]: Mike went to Paris. [X2]: Paris	[X1] [X2] is a [Y]	Yes No ...
Text Generation	Summarization	Las Vegas police ...	[X] TL;DR: [Y]	The victim ... A woman
	Translation	Je vous aime.	French [X] English: [Y]	I love you. I fancy you. ...

ID	Modifications	Acc. Gain
P413	<i>x plays in</i> → <i>at y position</i>	+23.2
P495	<i>x was created</i> → <i>made in y</i>	+10.8
P495	<i>x was</i> → <i>is created in y</i>	+10.0
P361	<i>x is a part of y</i>	+2.7
P413	<i>x plays in</i> → <i>y position</i>	+2.2

Table 6: Small modifications ([update](#), [insert](#), and [delete](#)) in paraphrase lead to large accuracy gain (%).

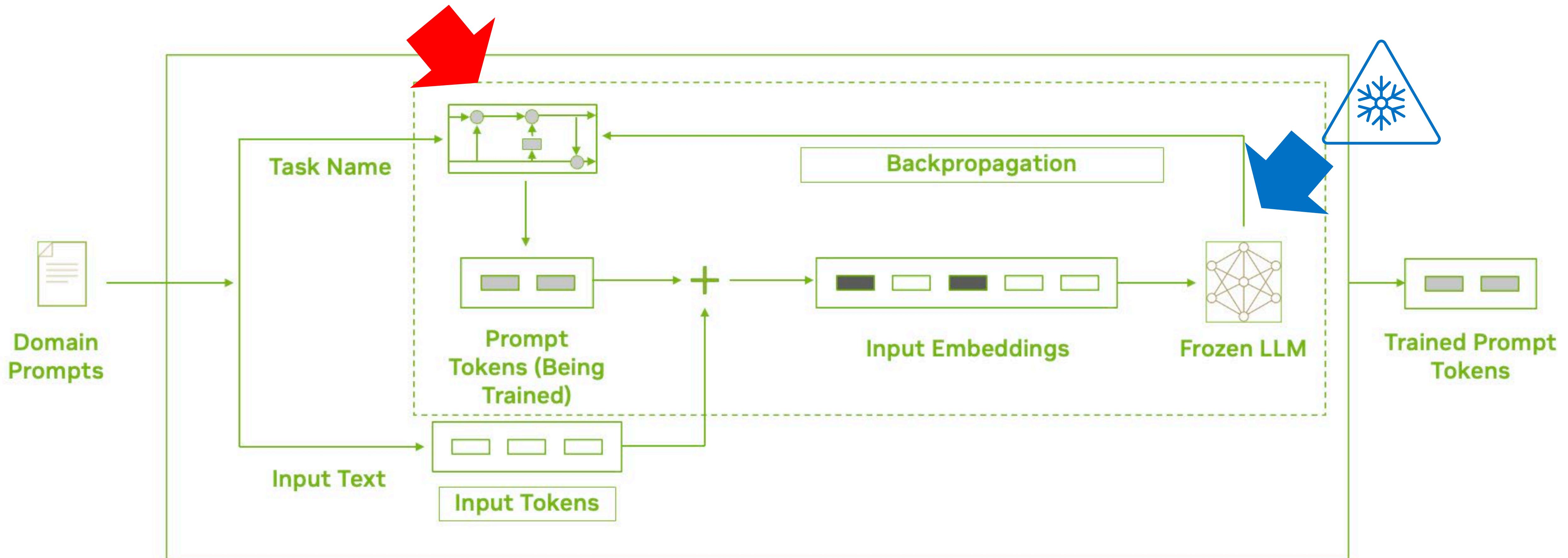
[Zhengbao Jiang et al. "How Can We Know What Language Models Know?".2020.](#)

Downstream Tasks

Customize LLM using prompt learning

Provide Context to Models

- Single model can serve many use-cases
- Achieve high accuracy for use-cases with just 100s of samples in minutes to hours
- Freeze foundational model, and learn the prompt tokens using a supervised learning approach



Downstream Tasks

Customize Models using Parameter-efficient tuning | Adapters

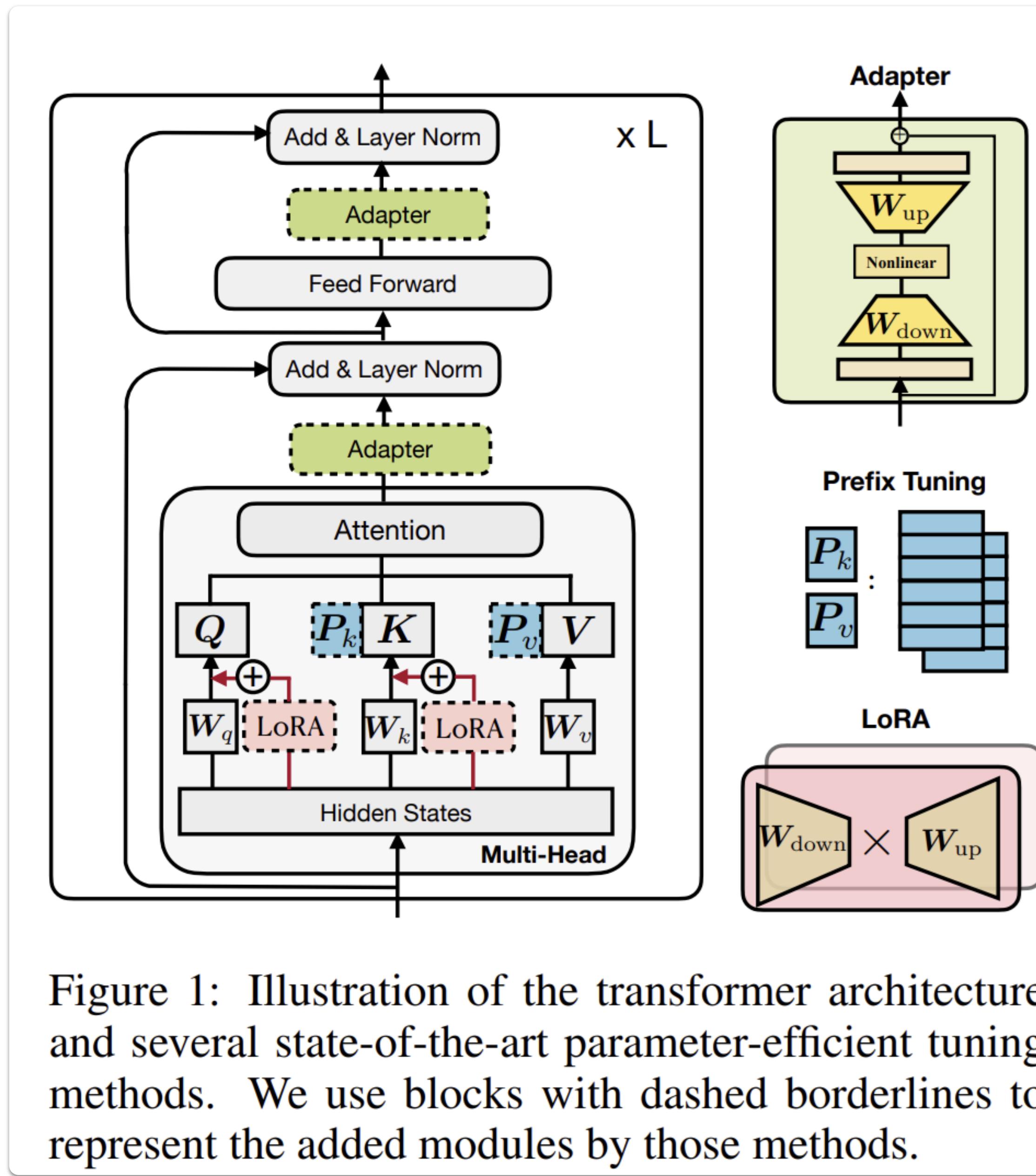


Figure 1: Illustration of the transformer architecture and several state-of-the-art parameter-efficient tuning methods. We use blocks with dashed borderlines to represent the added modules by those methods.

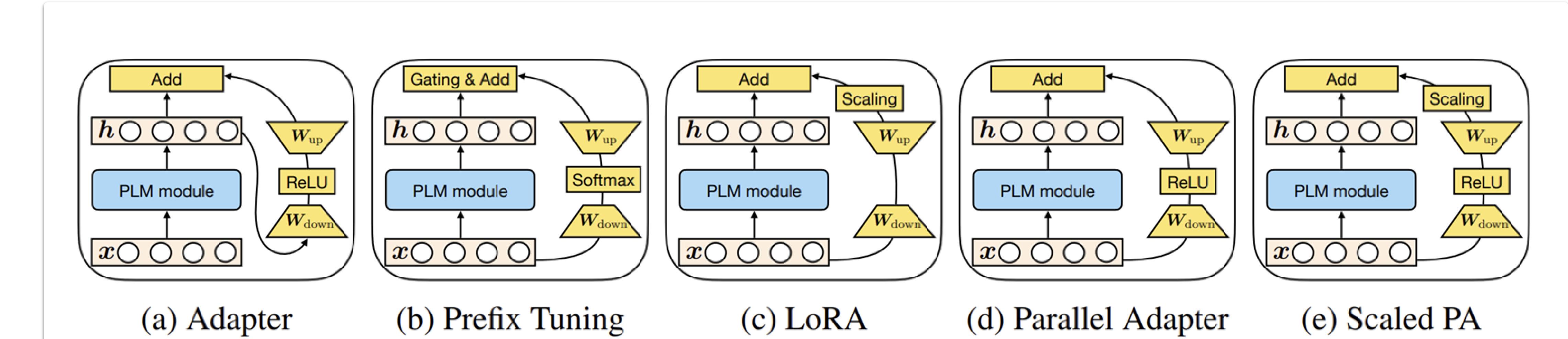
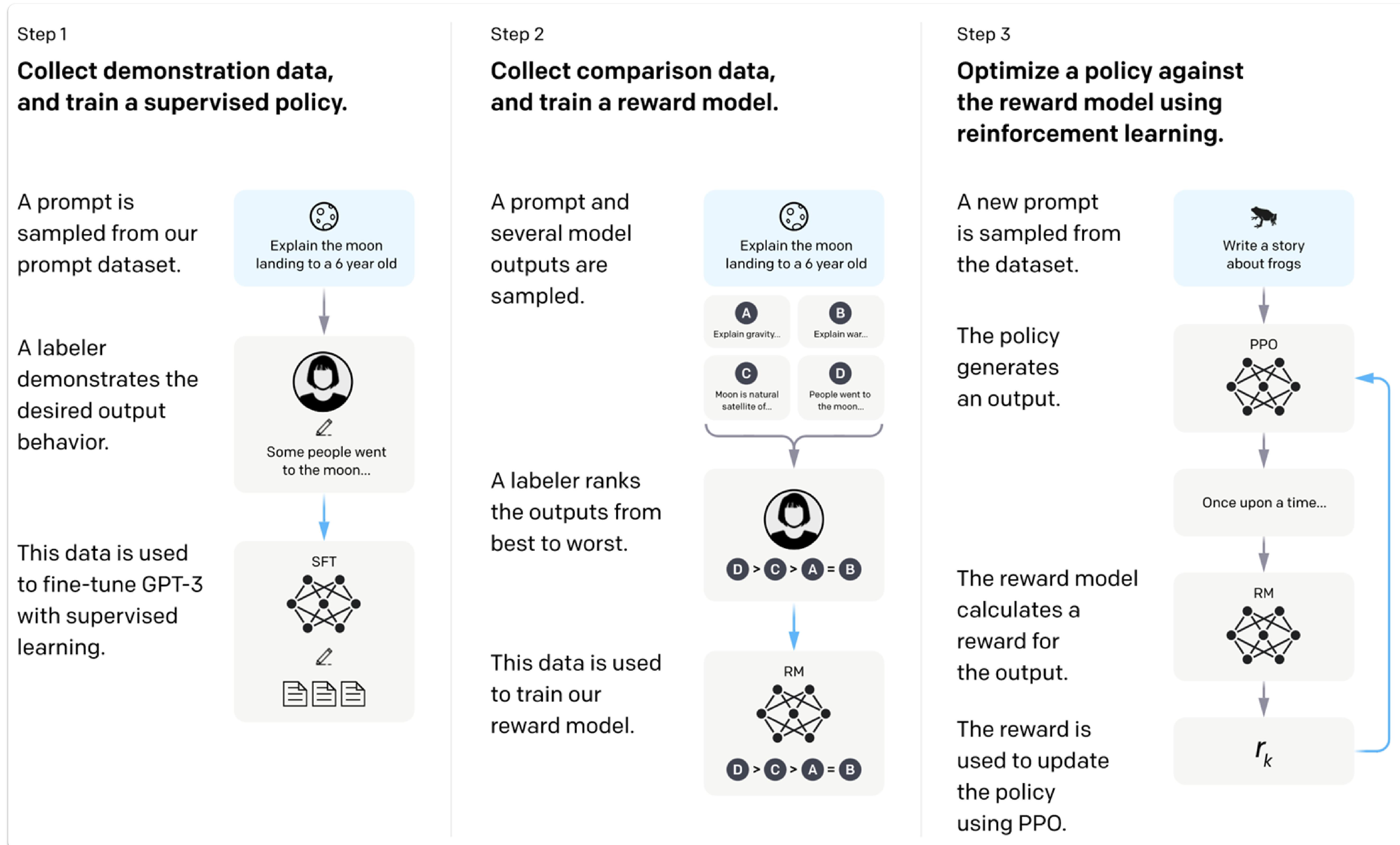
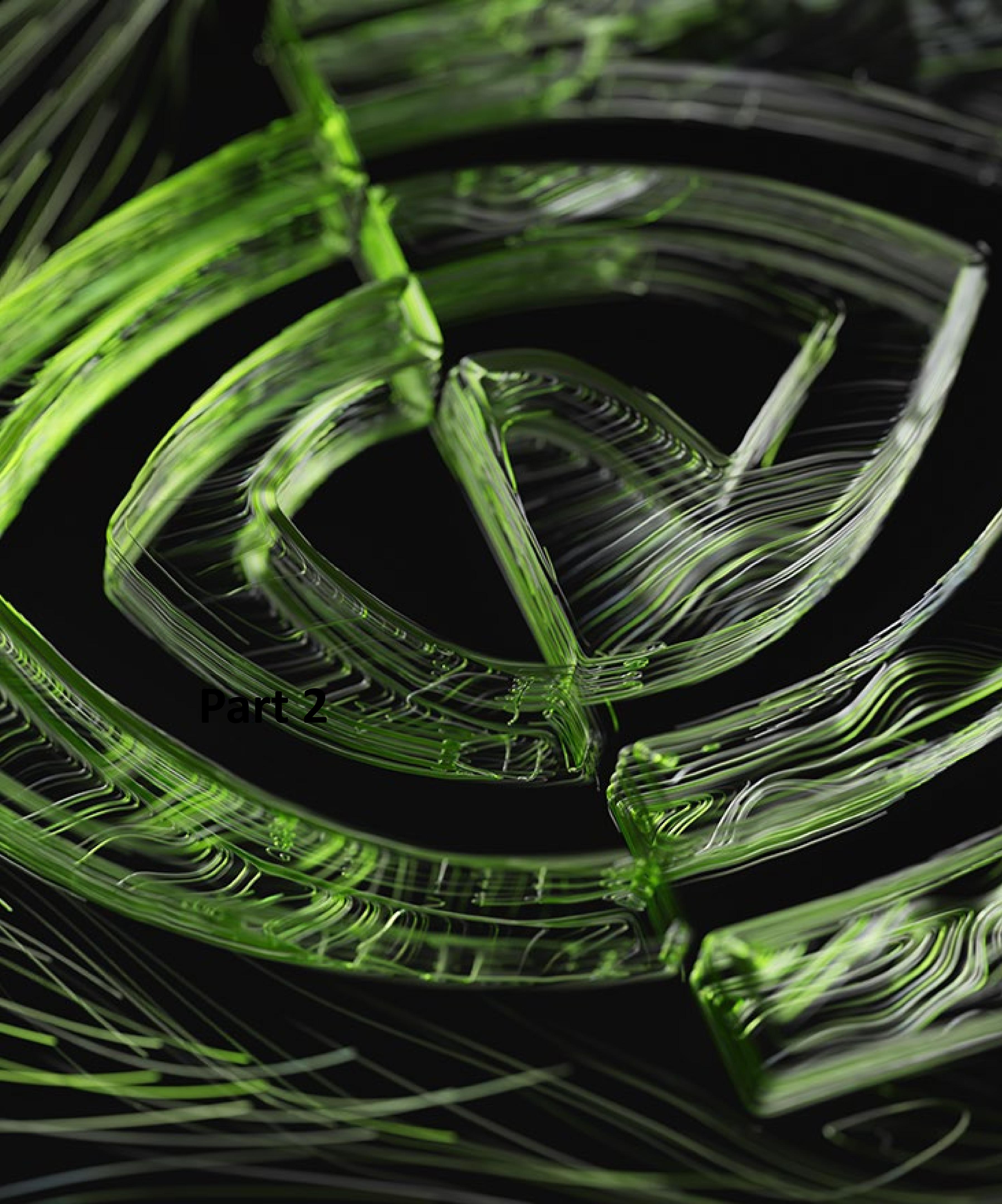


Figure 3: Graphical illustration of existing methods and the proposed variants. “PLM module” represents a certain sublayer of the PLM (e.g. attention or FFN) that is frozen. “Scaled PA” denotes scaled parallel adapter. We do not include multi-head parallel adapter here to save space.

Downstream Tasks

Customize LLM using user feedback





TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

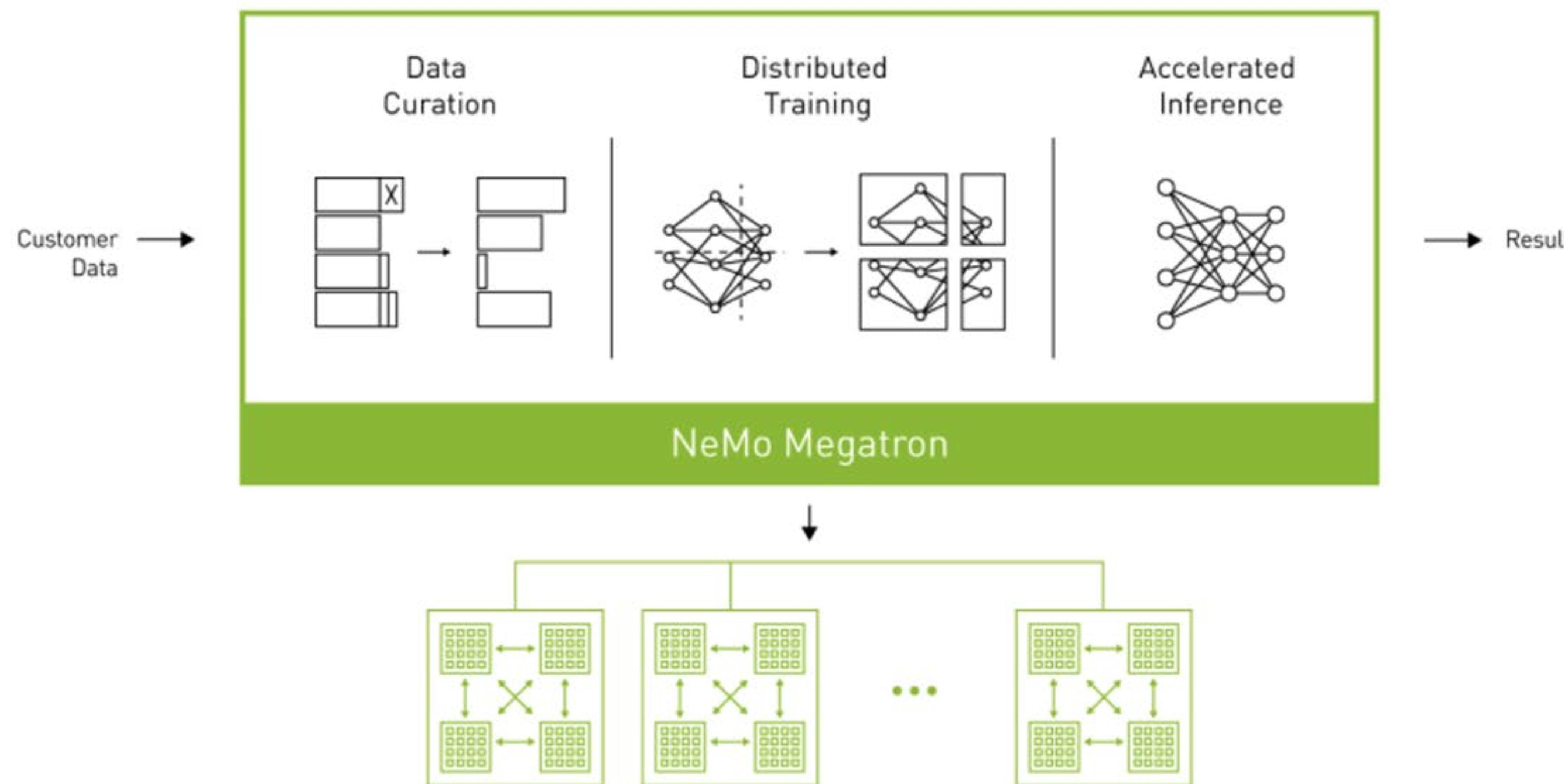
Part 2: Lab

- ASR, TTS and NLP Customization

NVIDIA NeMo Megatron

NVIDIA NeMo Megatron is an end-to-end framework for training and deploying LLMs with billions and trillions of parameters.

[Download Now](#)



developer.nvidia.com/nemo-megatron-open-beta



Downstream Tasks

NVIDIA NeMo

Prompt Learning

Within NeMo we refer to **p-tuning** and **prompt tuning** methods collectively as prompt learning. Both methods are parameter efficient alternatives to fine-tuning pretrained language models. Our NeMo implementation makes it possible to use one pretrained GPT model on many downstream tasks without needing to tune the model's full set of parameters. It also allows for adding new tasks to your model without overwriting or disrupting previous tasks for which the model has already been p-tuned/prompt-tuned. Because the original model parameters are frozen and never altered by either method, p-tuning/prompt-tuning also avoid cartographic forgetting issues often encountered when fine-tuning models.

Instead of selecting discrete text prompts in a manual or automated fashion, prompt tuning and p-tuning utilize virtual prompt embeddings that can be optimized via gradient decent. The only difference between prompt tuning and p-tuning within NeMo-Megatron is the architecture used to tune the soft prompt tokens during training.

- Our prompt tuning implementation is based off Lester et. al's EMNLP 2021 paper "[The Power of Scale for Parameter-Efficient Prompt Tuning](#)"
- Our p-tuning implementation is based off Liu et al's paper "[GPT Understands, Too](#)"

Adapters

In NeMo, we often train models and fine-tune them for a specific task. This is a reasonable approach when the models are just a few million parameters. However, this approach quickly becomes infeasible when approaching hundreds of millions or even billions of parameters. As a potential solution to such a scenario, where fine-tuning a massive model is no longer feasible, we look to [Adapters](#) [2] to specialize our model on a specific domain or task. Adapters require a fraction of the total number of parameters as the original model and are much more efficient to fine-tune.

Note

For a detailed tutorial on adding [Adapter](#) support to any PyTorch module, please refer to the [Tutorials for NeMo Adapters](#).

What are Adapters?

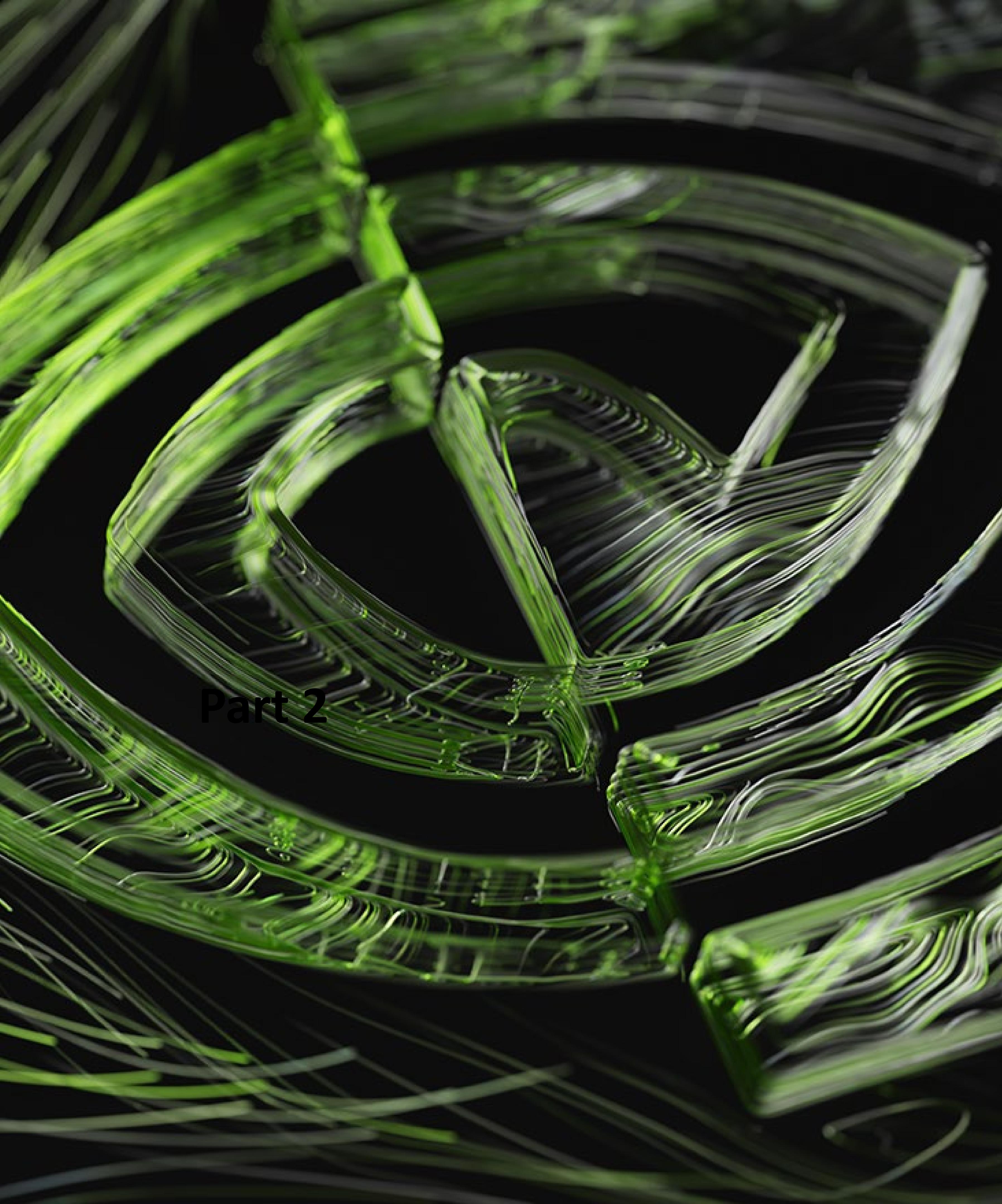
Adapters are a straightforward concept - one formulation can be shown by the diagram below. At their simplest, they are residual Feedforward layers that compress the input dimension (D) to a small

NVIDIA Riva NLU

Task	Architecture	Language	Dataset	Domain	Link
QA	BERT	English	SQuAD 2.0		Riva
QA	Megatron	English	SQuAD 2.0		Riva
Entity Recognition	BERT	English	GMB (Groningen Meaning Bank)	LOC, ORG, PER, GPE, TIME, MISC, O	Riva
Punctuation/Capitalization	BERT	English	Tatoeba sentences, Books from the Project Gutenberg, Transcripts from Fisher English Training Speech		Riva
Intent Detection & Slot Tagging	BERT	English	Proprietary	Weather	Riva
Intent Detection & Slot Tagging	DistilBERT	English	Proprietary	Misty (weather, smalltalk, places of interest)	Riva
Text Classification	BERT	English	Proprietary		Riva

NVIDIA Riva NMT

Model	Architecture	Source Language(s)	Target Language(s)	NGC link
mnmt_deesfr_en_transformer12x2	Transformer 12x2	German (de), Spanish (es), French (fr)	English (en)	NMT Multilingual De/Es/Fr En Transformer12x2
mnmt_en_deesfr_transformer12x2	Transformer 12x2	English (en)	German (de), Spanish (es), French (fr)	NMT Multilingual En De/Es/Fr Transformer12x2
mnmt_deesfr_en_transformer24x6	Transformer 24x6	German (de), Spanish (es), French (fr)	English (en)	NMT Multilingual De/Es/Fr En Transformer24x6
mnmt_en_deesfr_transformer24x6	Transformer 24x6	English (en)	German (de), Spanish (es), French (fr)	NMT Multilingual En De/Es/Fr Transformer24x6
en_de_24x6	Transformer 24x6	English (en)	German (de)	NMT En - De Transformer24x6
en_es_24x6	Transformer 24x6	English (en)	Spanish (es)	NMT En Es Transformer24x6
en_fr_24x6	Transformer 24x6	English (en)	French (fr)	NMT En Fr Transformer24x6
en_ru_24x6	Transformer 24x6	English (en)	Russian (ru)	NMT En Ru Transformer24x6
en_zh_24x6	Transformer 24x6	English (en)	Simplified Chinese (zh)	NMT En Zh Transformer24x6
de_en_24x6	Transformer 24x6	German (de)	English (en)	NMT De En Transformer24x6
es_en_24x6	Transformer 24x6	Spanish (es)	English (en)	NMT Es En Transformer24x6
fr_en_24x6	Transformer 24x6	French (fr)	English (en)	NMT Fr En Transformer24x6
ru_en_24x6	Transformer 24x6	Russian (ru)	English (en)	NMT Ru En Transformer24x6
zh_en_24x6	Transformer 24x6	Simplified Chinese (zh)	English (en)	NMT Zh En Transformer24x6



TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager

- Lab overview

Part 2: Lab

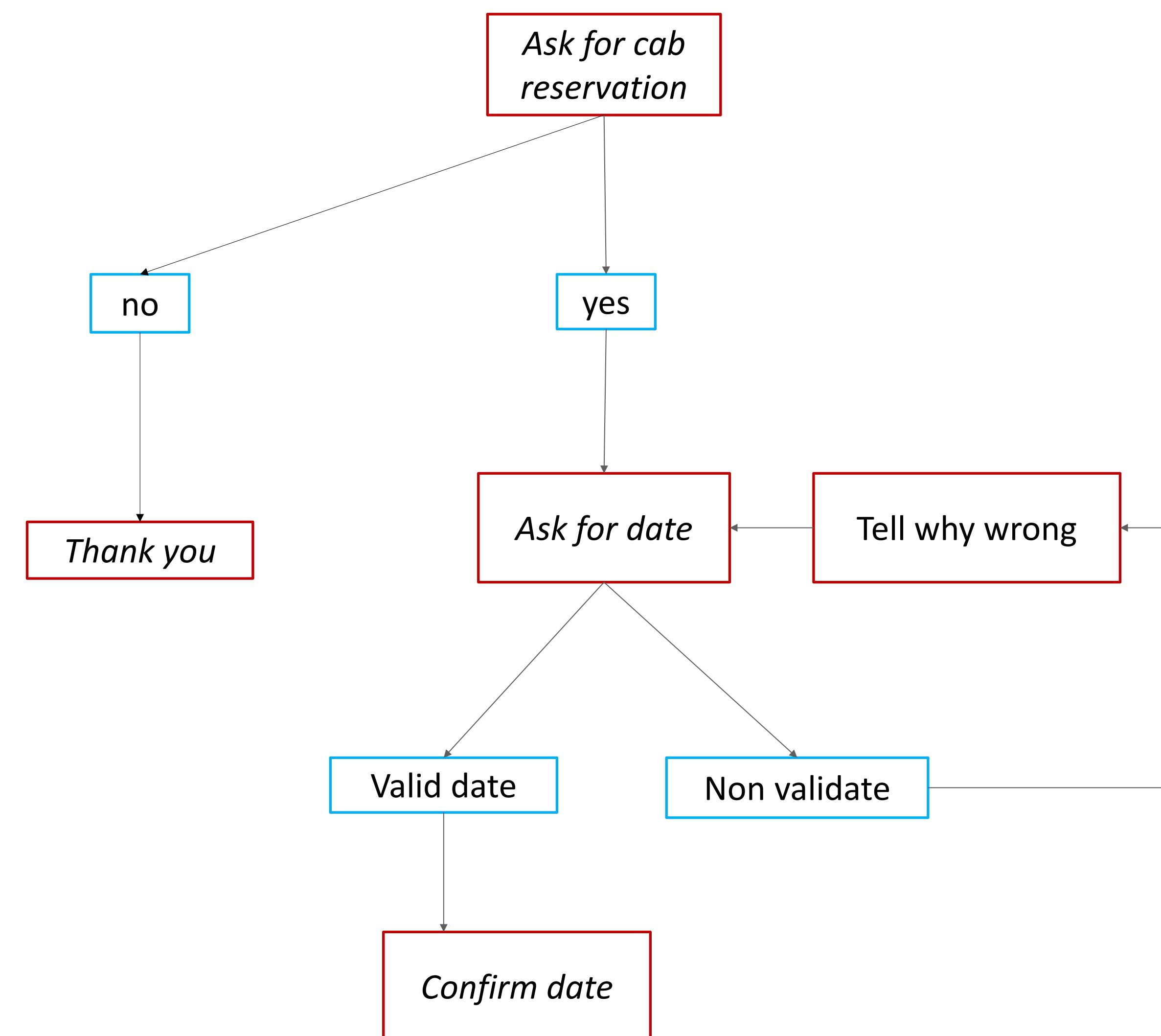
- ASR, TTS and NLP Customization

Dialog Manager

Types of dialogs

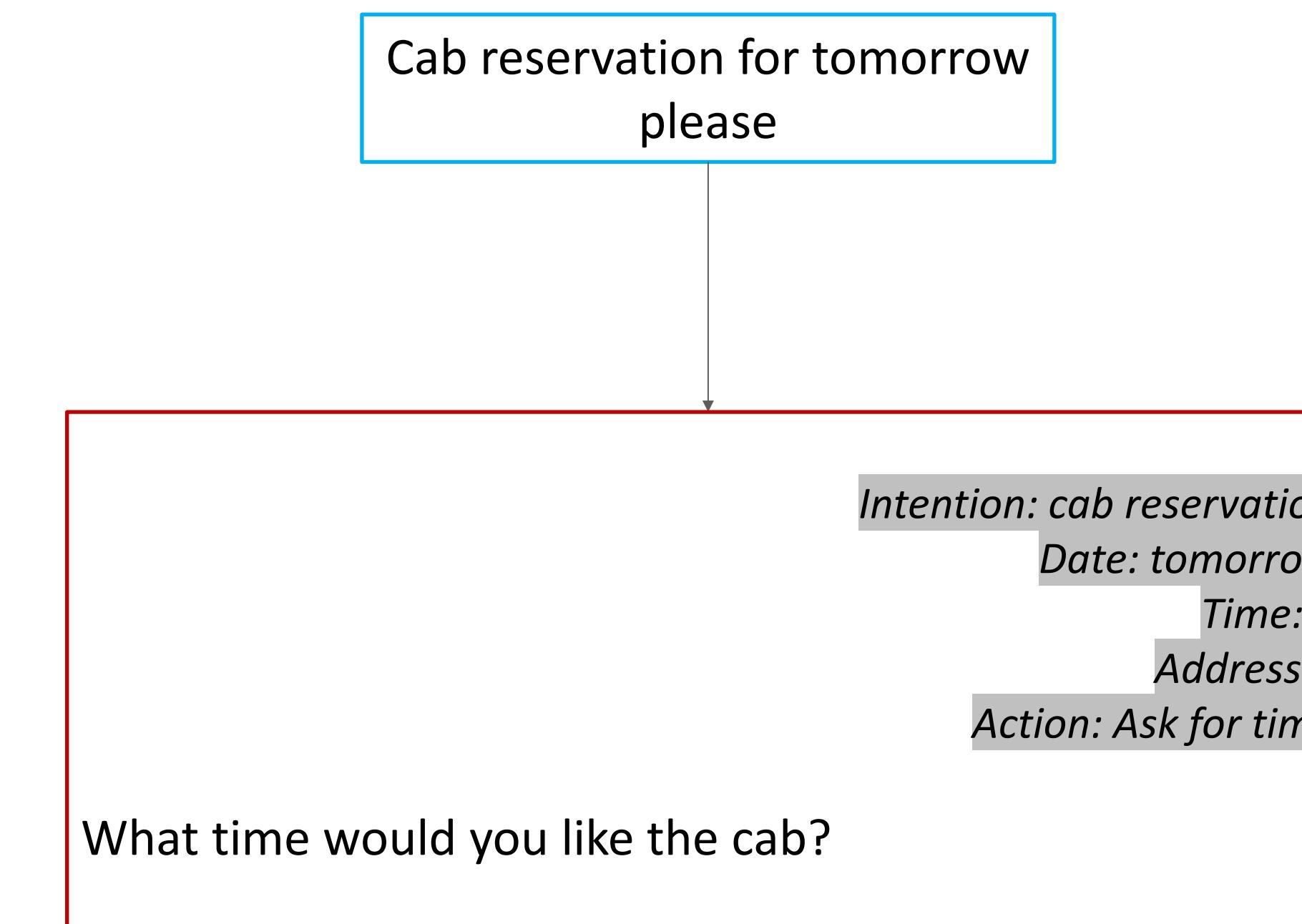
Finite State

Controlled



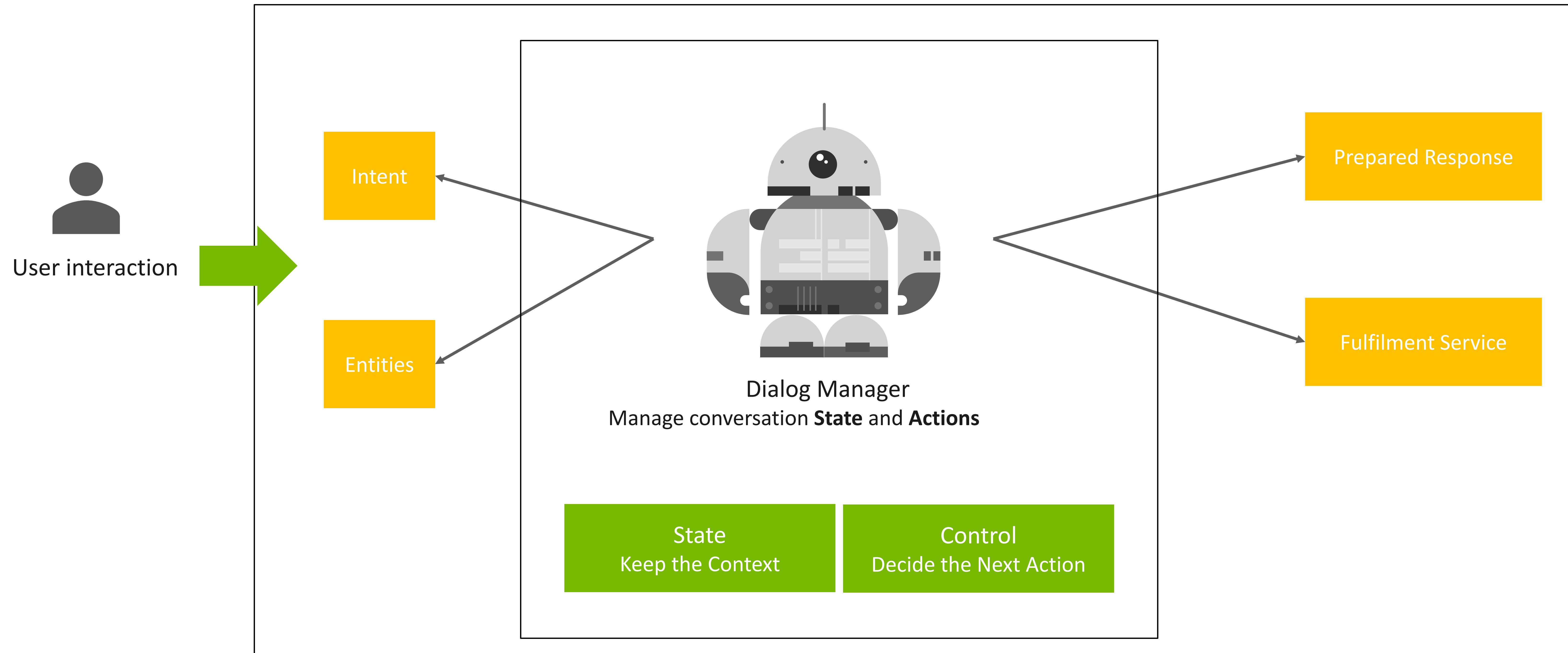
Form-Based

Identify data needed to act



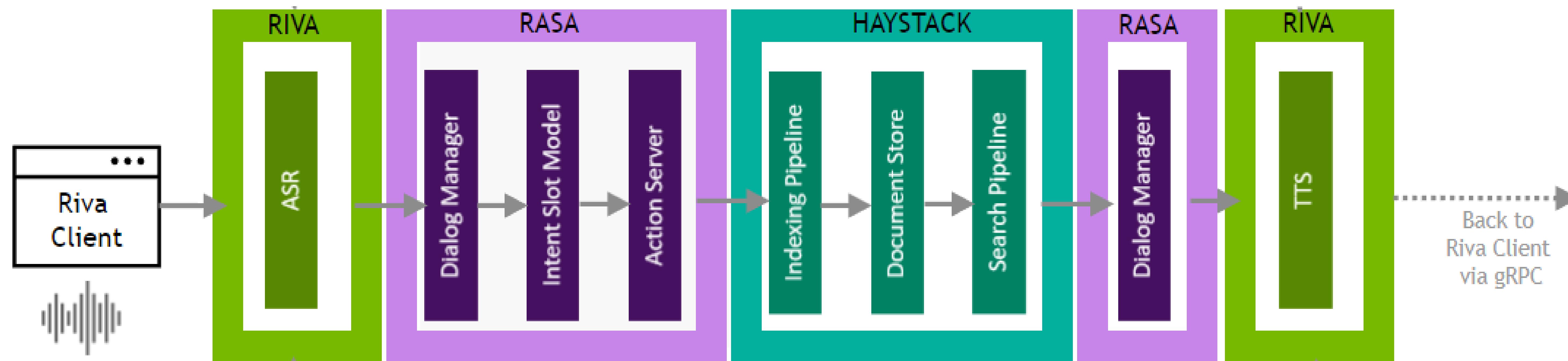
Dialog Manager

Architecture



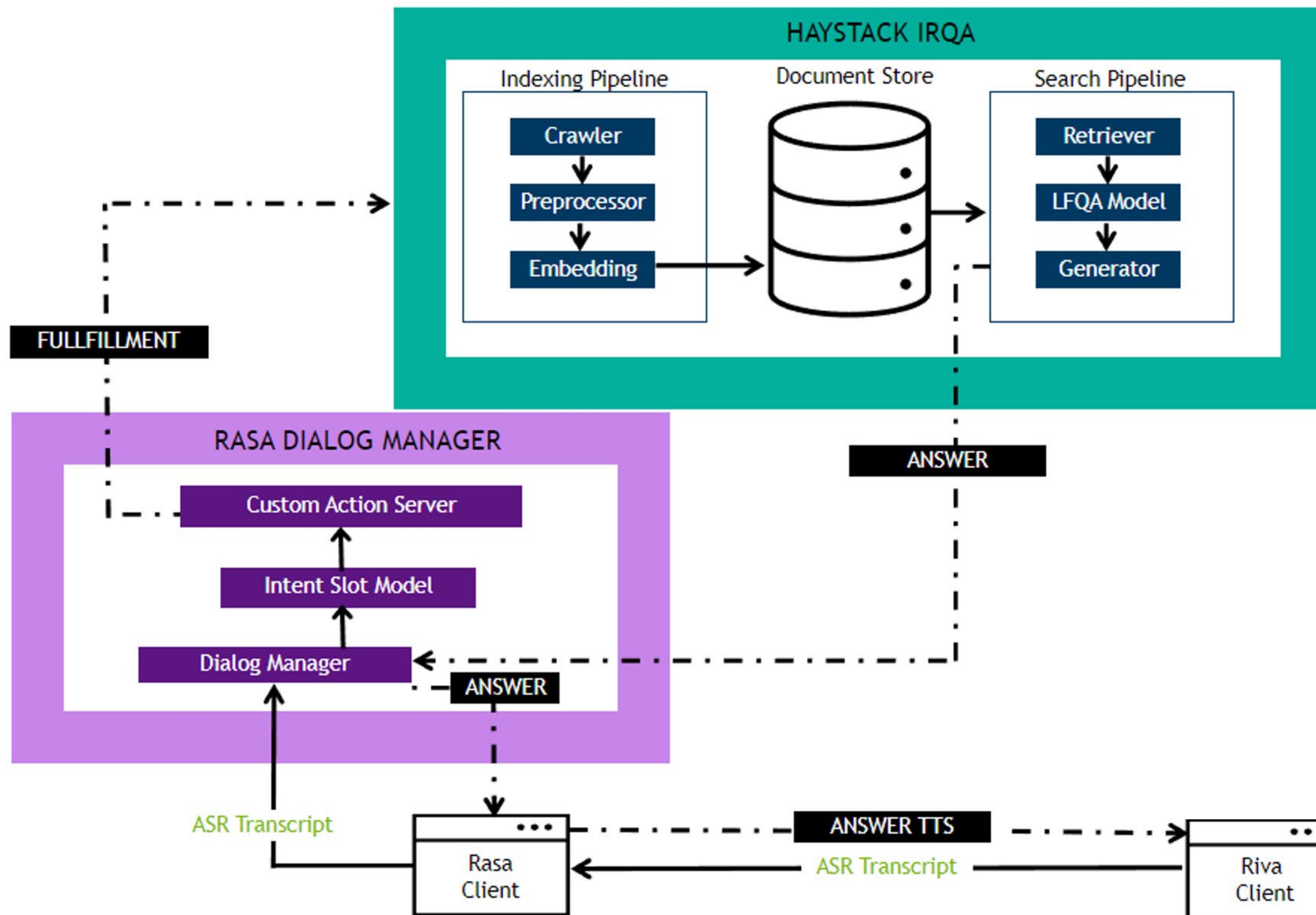
Dialog Manager

Virtual Assistant Example with Riva ASR and TTS, Rasa DM and Haystack IRQA



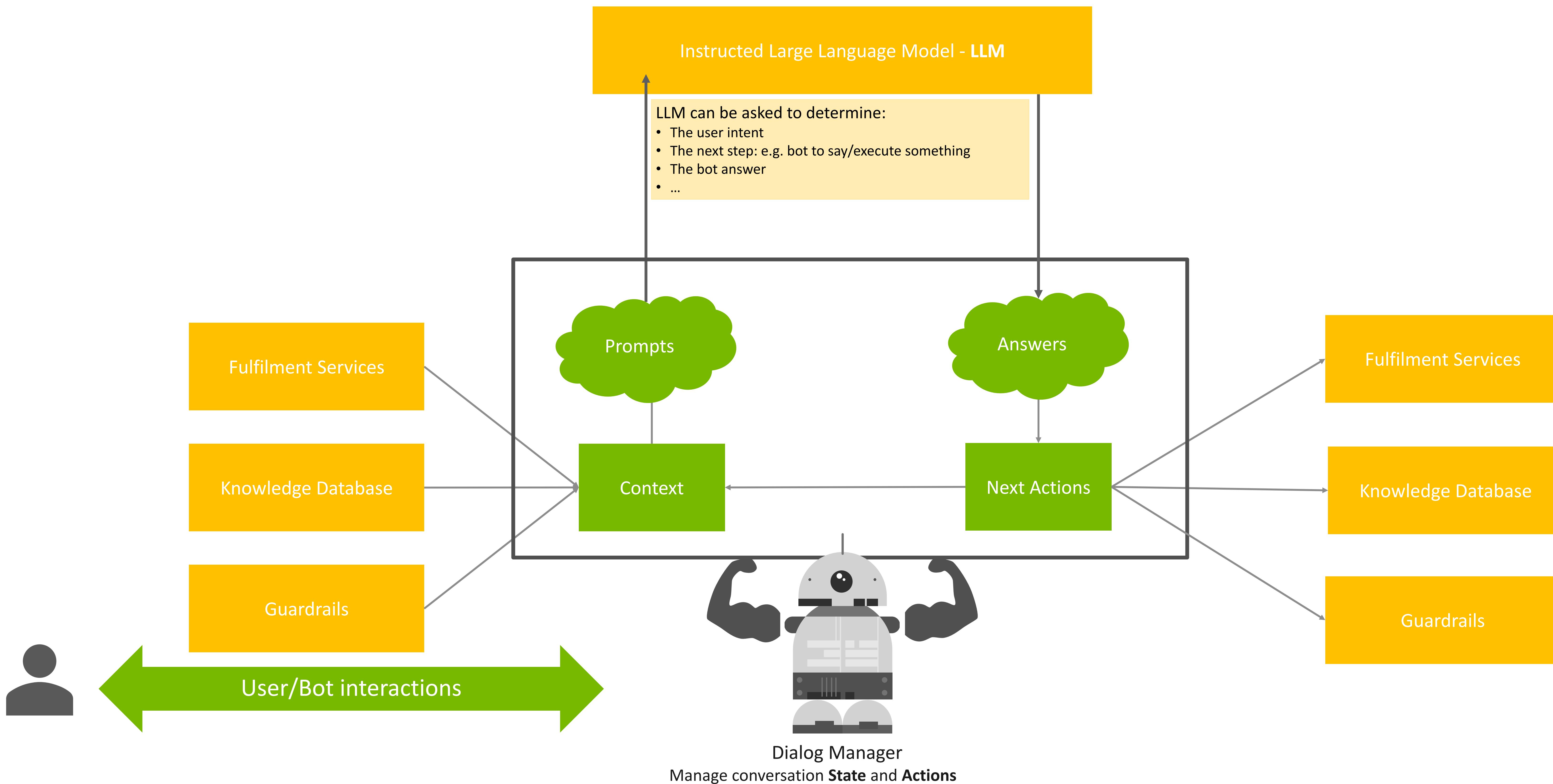
Dialog Manager

Virtual Assistant Example with Riva ASR and TTS, Rasa DM and Haystack IRQA



LLM-Powered Dialog Manager

Example of Architecture



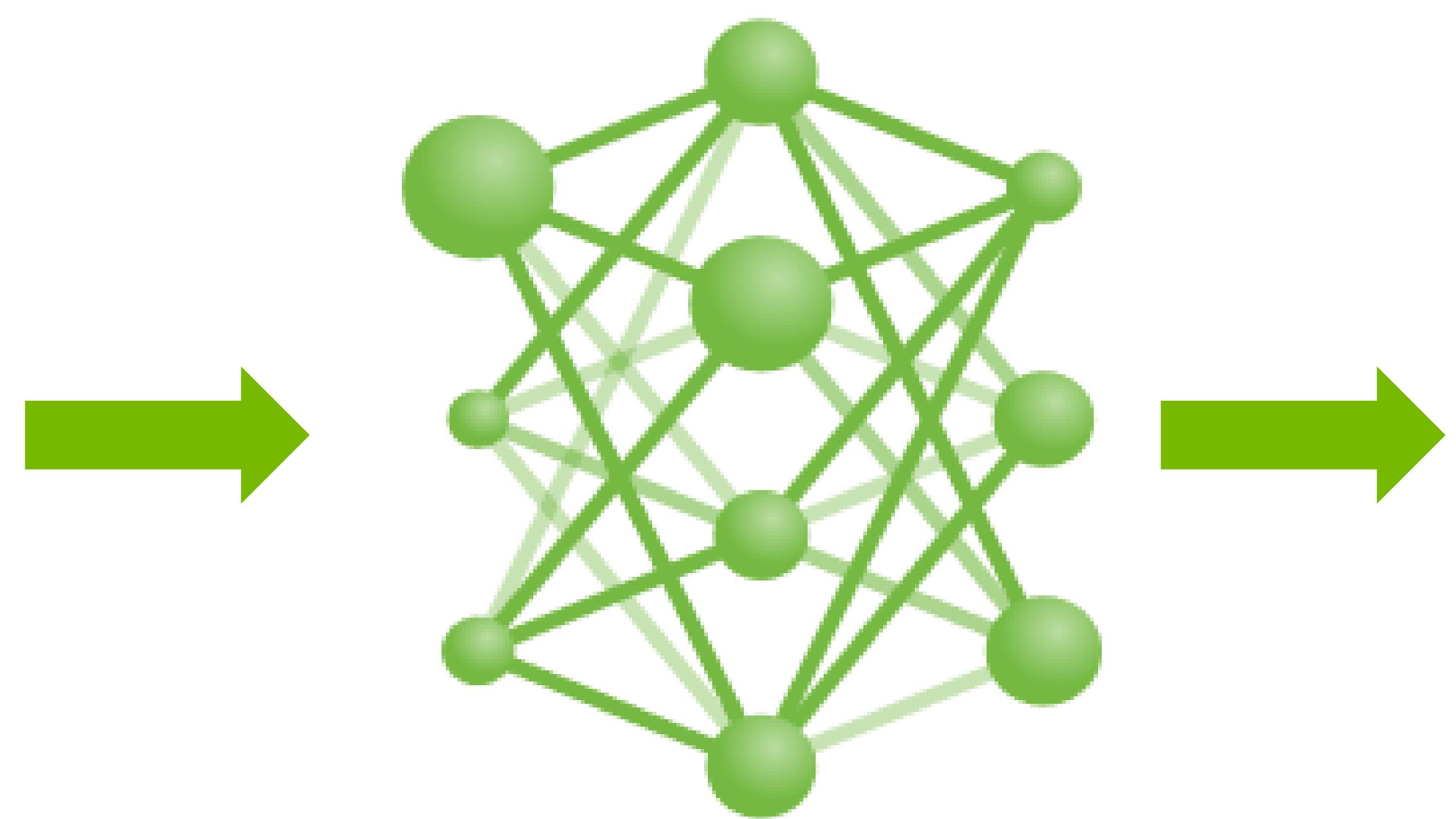
Instructed Large Language Model

Supervised LLM Finetuning to Follow Instructions and Dialogues

instruction: When did Virgin Australia start operating?

Context: Virgin Australia, the trading name of Virgin Australia Airlines Pty Ltd, is an Australian-based airline. It is the largest airline by fleet size to use the Virgin brand. It commenced services on 31 August 2000 as Virgin Blue, with two aircraft on a single route.^[3] It suddenly found itself as a major airline in Australia's domestic market after the collapse of Ansett Australia in September 2001. The airline has since grown to directly serve 32 cities in Australia, from hubs in Brisbane, Melbourne and Sydney.^[4]",

Response:



Virgin Australia commenced services on 31 August 2000 as Virgin Blue, with two aircraft on a single route.",

Instructed Large Language Model

Supervised LLM Finetuning to Follow Instructions and Dialogues

System: A chat between a curious user and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the user's questions.

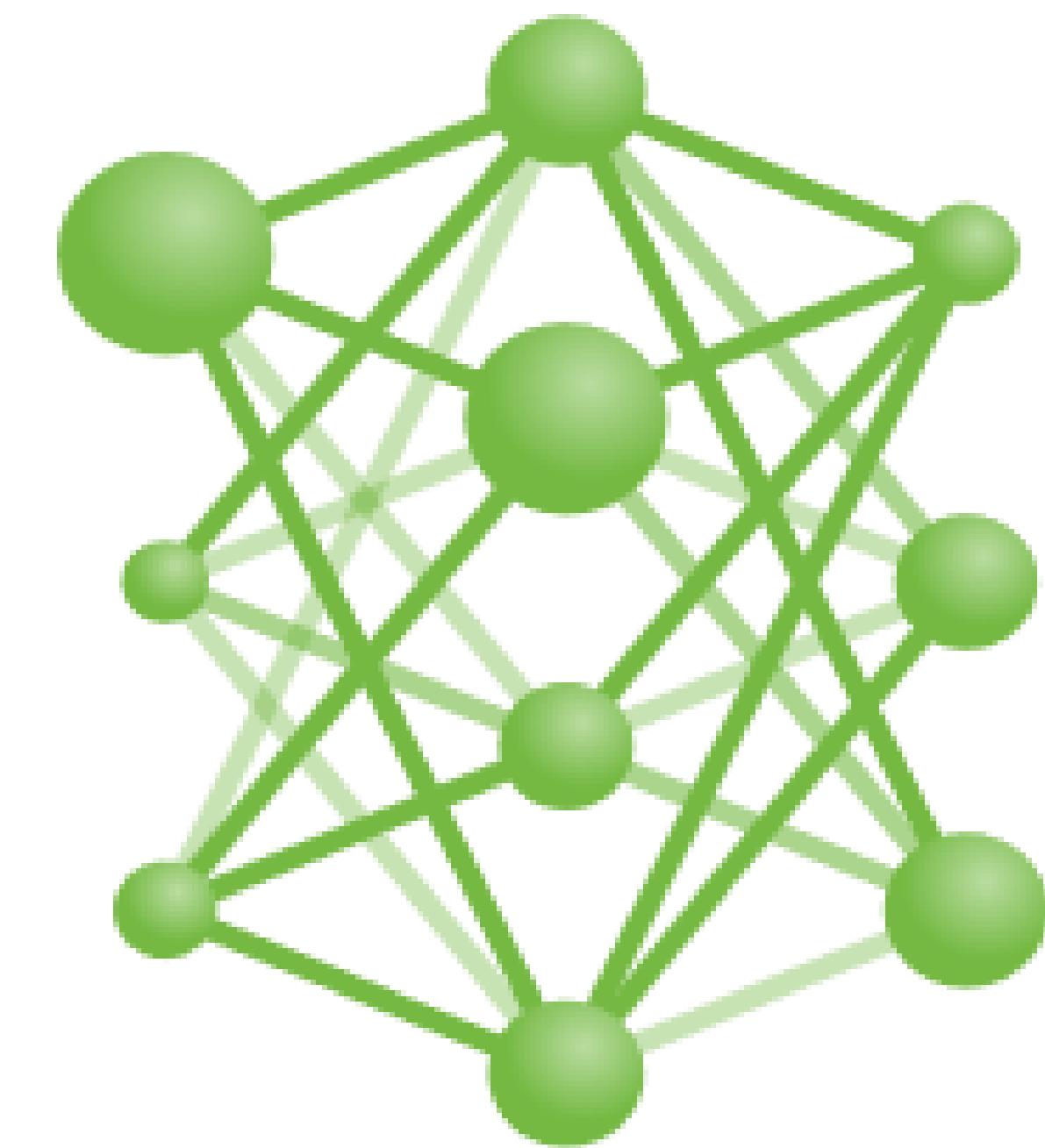
Conversations:

User: Who are you?

Assistant: I am NV Assistant, a language model trained by researchers from NVIDIA NeMo team.

User: What can you do?

Assistant:



I can chat with you

LLM-Powered Dialog Manager

Example: ChatGPT

ChatGPT

Free Research Preview: ChatGPT is optimized for dialogue. Our goal is to make AI systems more natural to interact with, and your feedback will help us improve our systems and make them even more useful.

Examples	Capabilities	Limitations
"Explain quantum computing in simple terms"	Remembers what user said earlier in the conversation	May occasionally generate incorrect information
"Got any creative ideas for a 10 year old's birthday?"	Allows user to provide follow-up corrections	May occasionally produce harmful instructions or biased content
"How do I make an HTTP request in Javascript?"	Trained to decline inappropriate requests	Limited knowledge of world and events after 2021

A Who is the president of the United States?

As of my knowledge cutoff of September 2021, the President of the United States was Joe Biden.

A Could you tell me where he born?

Certainly! Joe Biden was born in Scranton, Pennsylvania, United States.

A Could you tell me about his early career?

Yes, I'd be happy to give you a brief overview of Joe Biden's early career.

After graduating from law school in 1968, Biden worked as an attorney in Wilmington, Delaware, before being elected to the New Castle County Council in 1970. In 1972, at the age of 29, he was elected to the United States Senate, becoming one of the youngest senators in American history.

During his time in the Senate, Biden served on several committees, including the Foreign Relations Committee and the Judiciary Committee, and became known for his work on issues related to crime, drugs, and foreign policy. In 2008, he was elected Vice President of the United States under President Barack Obama, serving two terms in that role before being elected President in 2020.



TTS and Conversational AI Customization (Part 2)

Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Lab

- Investigate TTS Pipeline with NeMo and Riva
- ASR, TTS and NLP Customization

The Goal of the Lab

Beyond OOTB models

1. Build a virtual assistant agent with Riva ASR & TTS, and a simple finite state DM

2. Customize
 1. ASR: Word boosting, Lexicon Mapping
 2. TTS: pronunciations
 3. NLU: Named Entity Recognition for a Restaurant context

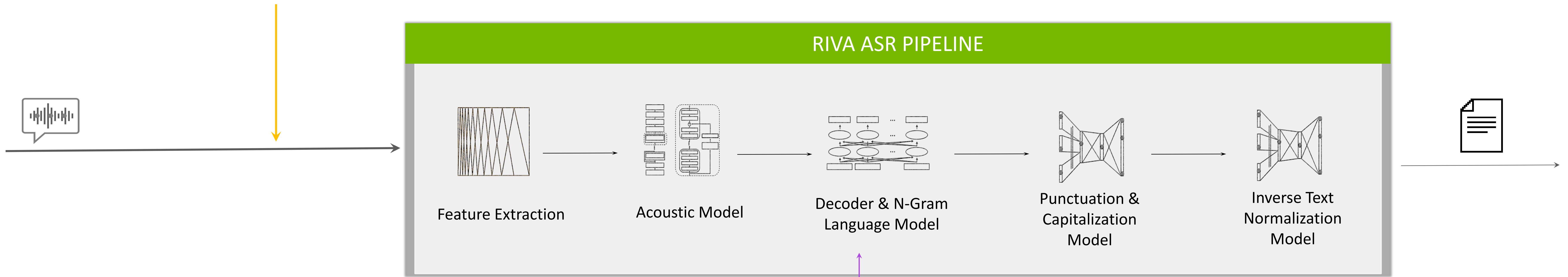


CUSTOMIZATIONS ACROSS RIVA ASR PIPELINE

Accurate | Customizable | Real-Time

Word Boosting* for:

- Product Names
- Jargon
- Context-Specific Knowledge



Adding Custom Pronunciation & Vocabulary

Vocabulary

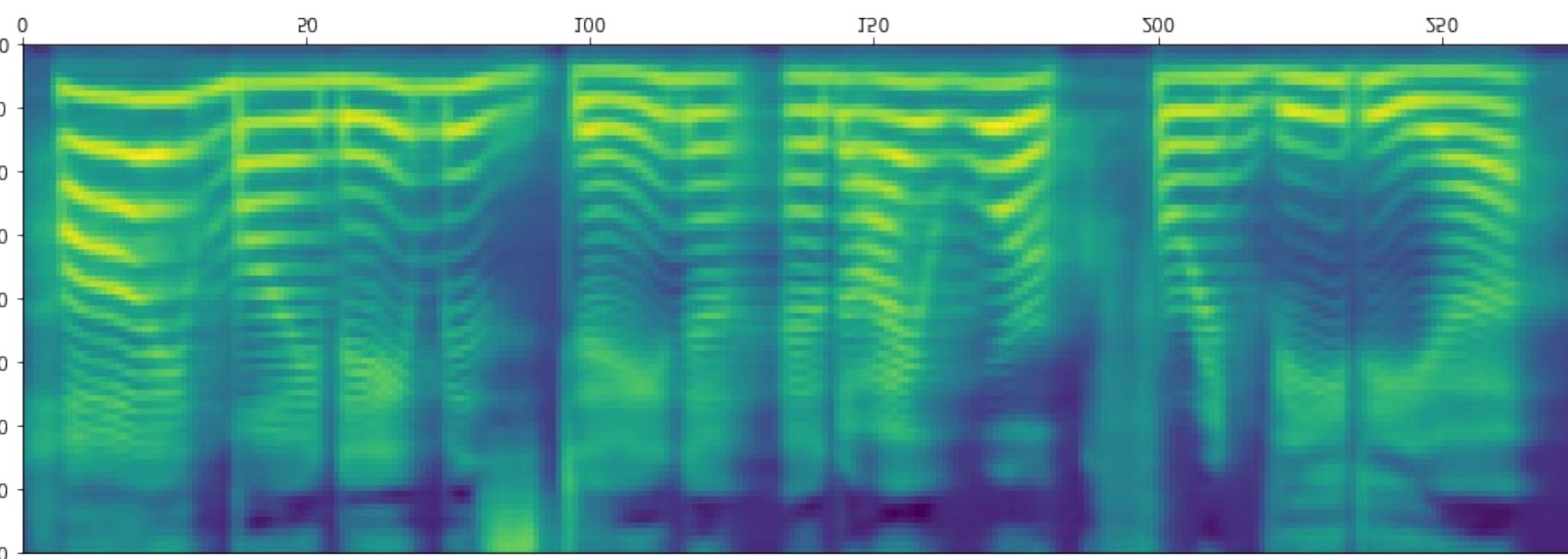
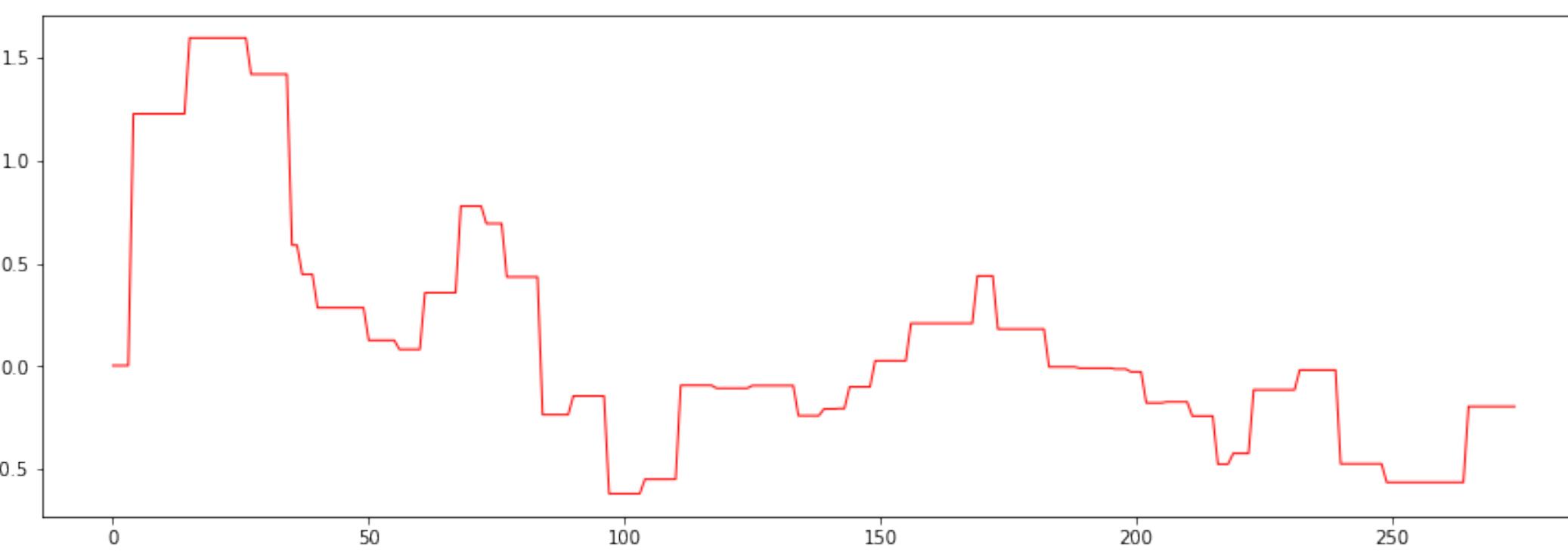
*the
i
to
and
a
...
emea*

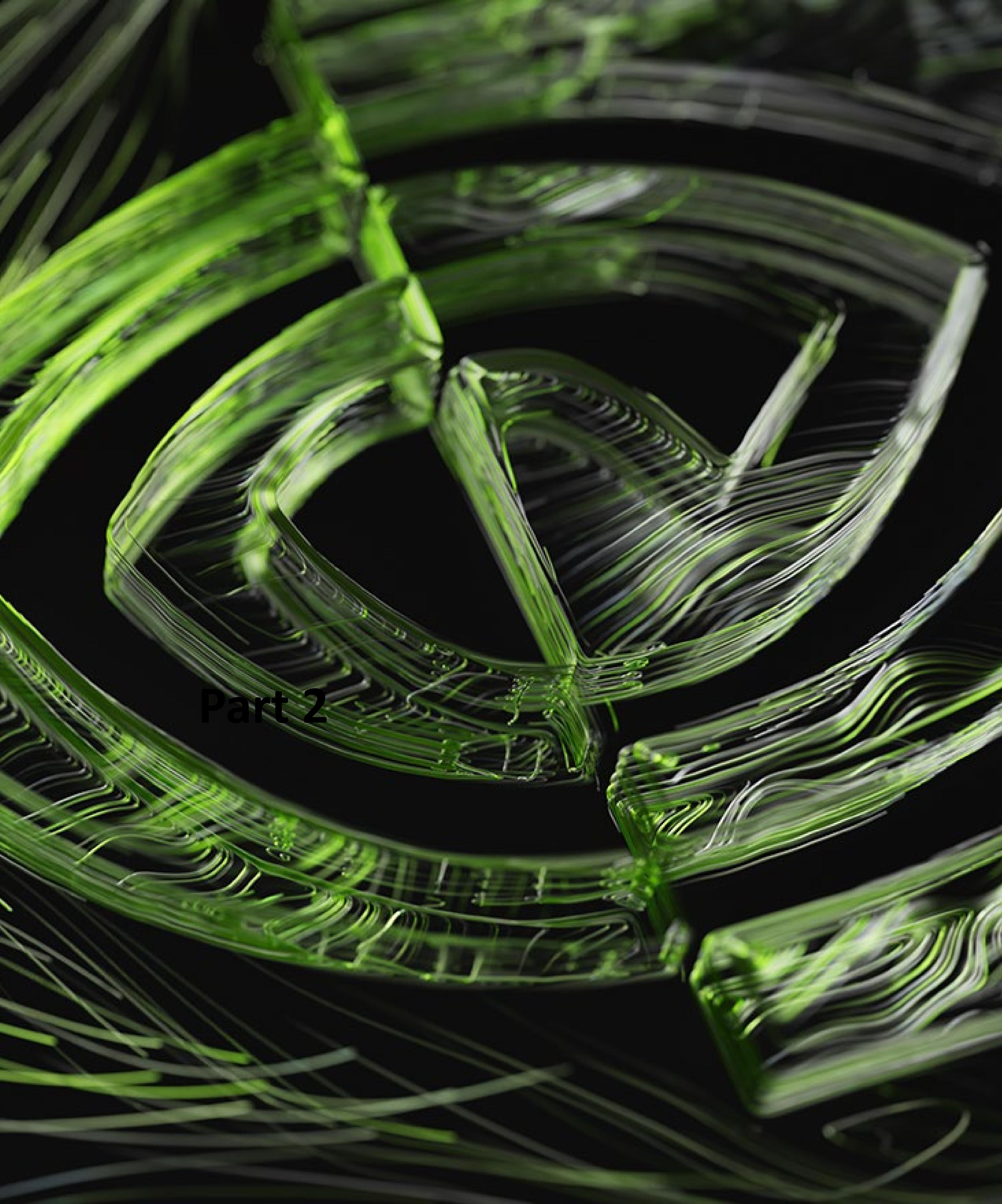
Lexicon

*with _with
not _not
this _this
my _my
as _as
don't _don 't
...
emea _e _me a
emea _ e _me a
emea _e _m e a*

NVIDIA Riva Text-To-Speech

HH	AY1	M	AY1	N	EY1	M	i	s	D	EY1	N	AH0	a	n	d	AY1	W	ER1	K	f	a	r	EHO	N	V	IY1	D	IY0	AH0										
7	41	97	0	11	41	0	12	50	11	0	78	88	0	3	50	12	31	0	70	83	73	0	41	0	21	47	9	0	75	84	87	0	43	12	20	56	3	55	31





TTS and Conversational AI Customization (Part 2)

Part 1: Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Lab overview

Part 1: Lab

- Investigate TTS Pipeline with NeMo and Riva

Part 2: Lecture

- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Part 2: Lab

- ASR, TTS and NLP Customization



TTS and Conversational AI Customization (Part 2)

Lecture

- Speech Synthesis
- TTS Tools: Riva, NGC, and NeMo
- Natural Language Understanding
- NLU Tools: Riva, NGC, and NeMo
- Dialog Manager
- Lab overview

Lab

- Investigate TTS Pipeline with NeMo and Riva
- ASR, TTS and NLP Customization

!

Part 2: Lab

