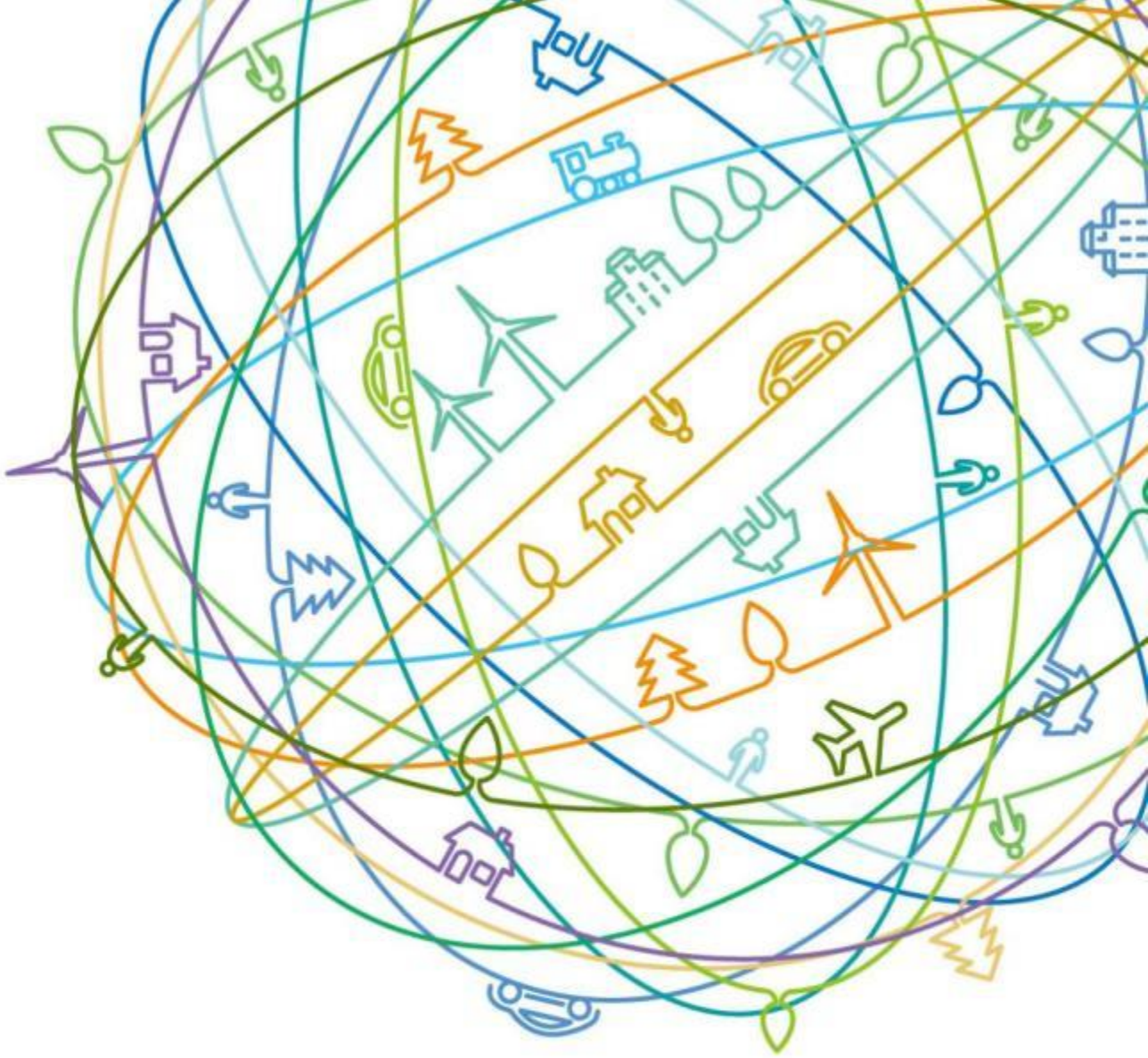
Key Technologies and Future Evolution of TTS



Reporter： Parkhomenko Denis

Time： 2020-12-05

Goal

Touch TTS methods, try to catch emerging trends and predict future evolution and future technology of AI speech domain.

Contents

What is TTS?

What is TTS?

Привет уважаемые слушатели! меня зовут Ольга. Наверное, вы поняли что я - голосовой движок Хуавэй, и я могу говорить на десяти языках.

Bonjour chers auditeurs! je m'appelle Celia. vous avez probablement compris que je suis le moteur vocal de Huawei et que je peux parler dix langues!

Hola queridos oyentes! Mi nombre es Celia. ¡probablemente entendiste que soy el motor de voz de Huawei y puedo hablar diez idiomas!

مرحبا اعزائي المستمعين! اسمي سيليا. ربما فهمت أنني محرك صوت Huawei ويمكن أن أتكلم عشر لغات!

What is TTS? Applications?

[6]

Any text to produce!

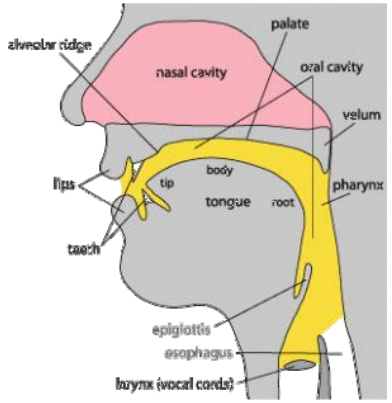
⇩

Text-to-Speech method

⇩

Intelligible speech

In 1779 the German scientist Christian Gottlieb Kratzenstein won the first prize in a competition announced by the Russian Imperial Academy of Sciences and Arts for models he built of the human vocal tract that could produce the five long vowel sounds. In International Phonetic Alphabet notation:
[aː], [eː], [iː], [oː] and [uː].



What is TTS? Applications?

[5]

Organization	Scientists	Challenges
ISCA Internatioonal Speech Communication Association	Junichi Yamagishi (chairman ISCA)	Blizzard (Organizer)
Austrian academy of sciences		Blizzard (sponsor)
University of Science and Technology of China	Li-Rong Dai Zhen-Hua Ling	Blizzard best in 2019
Beijing Lingban Technology Co. Ltd	Fengyun Zhu Yansuo Yu	Blizzard top 4 2019
Guangzhou Deepsound Technology Co. Ltd	Boxian Huang Xu Wang	Blizzard top 4 2019
Tencent Technology Co., Ltd	Shan Liu Jing Chen	Blizzard top 4 2018
University of the Basque Country		Voice Conversion Challenge
STMS-IRCAM/Sorbonne University		Voice Conversion Challenge
Academia Sinica		Voice Conversion Challenge

Voice assistants:
- Huawei (Silia),
- Yandex (Alice),
- Apple (Siri),
- Google,
- Amazon (Alexa),
- Microsoft (Cortana),
- Mail.ru (Marusia) …

Audio books, AAC devices,
Phone banking, Call service,
Smart TV, Gaming consoles,
Language learning, Smart Toys,
Navigation, Transportation…

and many others fields…

What is TTS? Applications?

[3,4]

Amazon TTS coverage

Language	Language (Area)	Language Code
arabic	Arabic	arb
Chinese language	Chinese, Mandarin	cmnCN
danish	Danish	daDK
Netherlands language	Dutch	nlNL
engl	English, Australian	enAU
	English, British	enGB
	English, Indian	enIN
	English, US	enUS
	English, Welsh	enGBWLS
french	French	frFR
	French, Canadian	frCA
hindi	Hindi	hiIN
German	German	deDE
Icelandic	Icelandic	isIS
italian	Italian	itIT
Japanese	Japanese	jaJP
korean	Korean	koKR
norwegian	Norwegian	nbNO
Polish	Polish	plPL
portuguese	Portuguese, Brazilian	ptBR
	Portuguese, European	ptPT
romanian	Romanian	roRO
russian	Russian	ruRU
spanish	Spanish, European	esES
	Spanish, Mexican	esMX
	Spanish, US	esUS
swedish	Swedish	svSE
turkish	Turkish	trTR
Wales language	Welsh	cyGB
Total languages: 21	**Total dialects: 29**	

Google TTS coverage

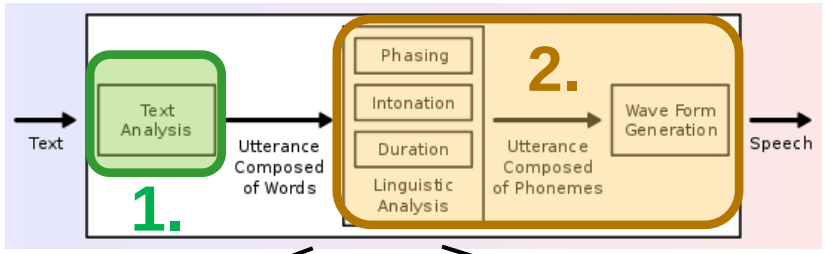
Language	Language (region)	Number of voices
Polish	Polish (Poland)	5
danish	Danish (Denmark)	1
German	German (Germany)	2
russian	Russian (Russia)	4
french	French (France)	4
	French (Canada)	4
korean	Korean (Korea)	4
Netherlands language	Netherlands (Netherlands)	1
norwegian	Norwegian (Norway)	1
portuguese	Portuguese (Brazil)	1
	Portuguese (Portugal)	4
Japanese	Japanese (Japan)	1
swedish	Swedish (Sweden)	1
slovak	Slovak (Slovakia)	1
turkish	Turkish (Turkey)	5
ukrainian	Ukrainian (Ukraine)	1
spanish	Spanish (Spain)	1
italian	Italian (Italy)	1
engl	English (Australia)	4
	English (United States)	4
	English (United Kingdom)	4
Total languages: 17	**Total dialects: 21**	**Total voices: 54**

Basics and conventional TTS technologies

Conventional TTS technologies

[7]

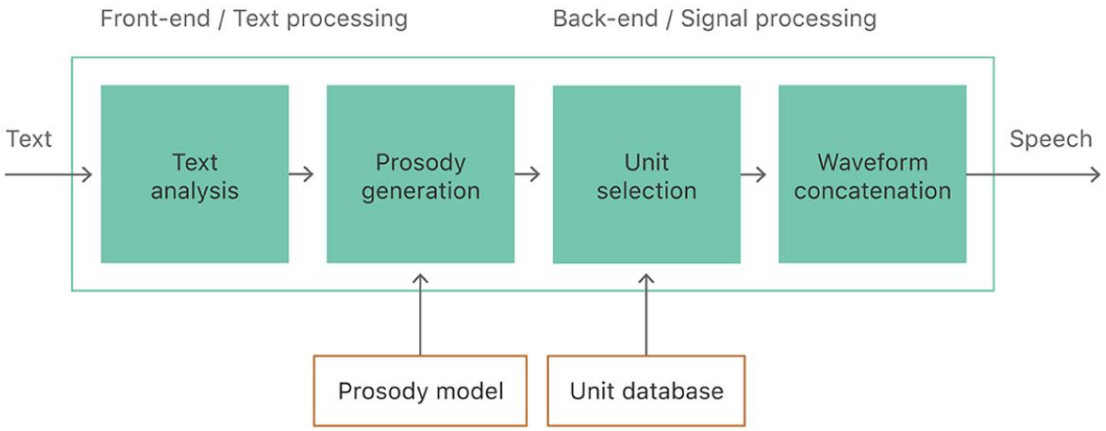
Overview of a typical TTS system



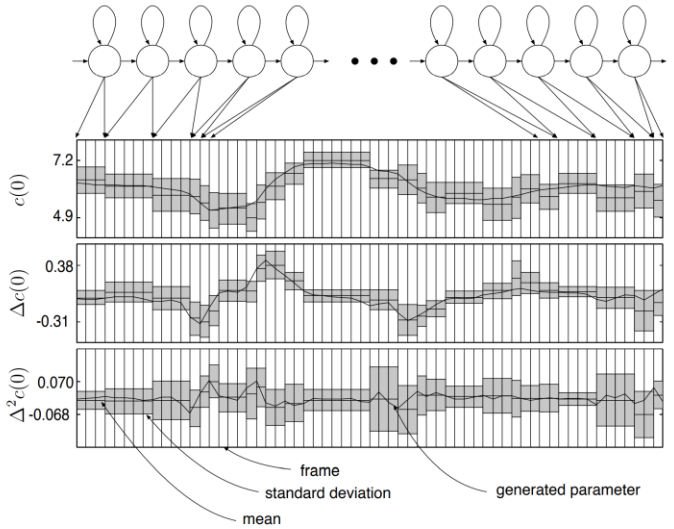
model describes those parameters using statistics (e.g., means and variances of probability density functions) which capture the distribution of parameter values found in the training data.

because model describes the speech using parameters, rather than stored exemplars.

Concatenative unit selection



Statistical parametric Synthesis



Pros:

Transforming voice characteristics, speaking styles, voice mixture, robust to train data noise/fluctuation.

Cons:

Speech quality is affected by vocoders, acoustic model accuracy and over-smoothing

Conventional TTS technologies

[8]

Front-End: Text preprocessing

1. **Diacritization**
   1. السلام عليكم
   2. السَّلامُ عَلَيْكُمْ
2. **Stress prediction**
   1. Катя мячик, Катя бежала.
   2. Катя’ мя’чик, Ка’тя бежа’ла.
3. **Normalization**
   1. She was born on 1994
   2. She was born on nineteen ninety four
4. **Phonemization** (IPA or Arpabet for English)
   1. She was born on 1994 –
   2. SH IY1 / W AA1 Z / B AO1 R N / AA1 N / W AH1 N / TH AW1 . Z AH0 N D / N AY1 N / HH AH1 N . D R AH0 D / AH0 N D / N AY1 N . T IY0 / F AO1 R
   3. 你好，你好吗？(nǐ hǎo nǐ hǎo mǎ)

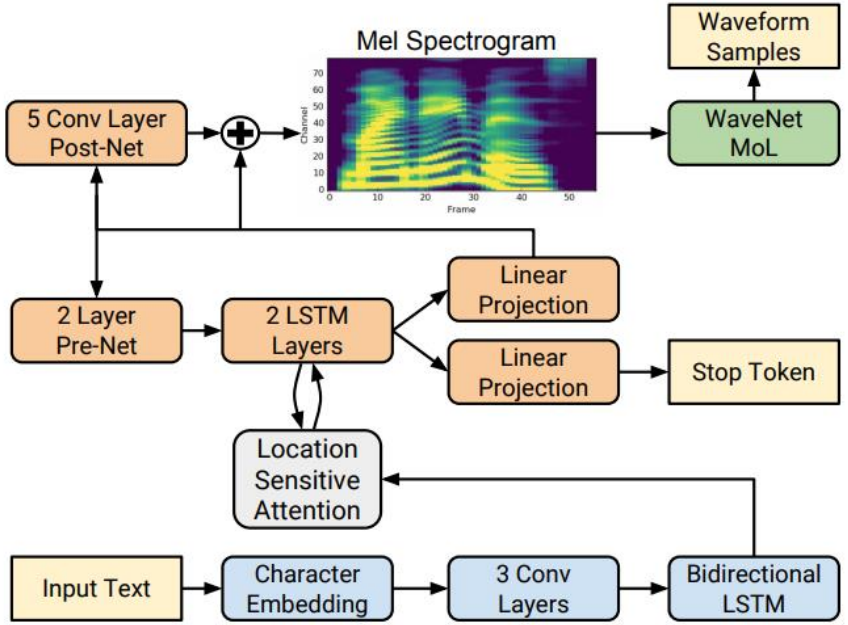
	EXPN	abbreviation	*adv, N.Y, mph, gov't*
alpha	LSEQ	letter sequence	*CIA, D.C, CDs*
	ASWD	read as word	*CAT*, proper names
	MSPL	misspelling	*geogaphy*
	NUM	number (cardinal)	*12, 45, 1/2, 0·6*
	NORD	number (ordinal)	*May 7, 3rd, Bill Gates III*
	NTEL	telephone (or part of)	*212 555-4523*
	NDIG	number as digits	*Room 101*
N	NIDE	identifier	*747, 386, I5, pc110, 3A*
U	NADDR	number as street address	*5000 Pennsylvania, 4523 Forbes*
M	NZIP	zip code or PO Box	*91020*
B	NTIME	a (compound) time	*3·20, 11:45*
E	NDATE	a (compound) date	*2/2/99, 14/03/87* (or US) *03/14/87*
R	NYER	year(s)	*1998, 80s, 1900s, 2003*
S	MONEY	money (US or other)	*$3·45, HK$300, Y20,000, $200K*
	BMONEY	money tr/m/billions	*$3·45 billion*
	PRCT	percentage	*75%, 3·4%*
	SPLT	mixed or "split"	*WS99, x220, 2-car* (see also SLNT and PUNC examples)
	SLNT	not spoken, word boundary or emphasis character:	
M		word boundary	*M.bath, KENT*RLTY, _really_*
I	PUNC	not spoken, non-standard punctuation: "***" in	
S		phrase boundary	*$99,9K***Whites*, "..." in *DECIDE...Year*
C	FNSP	funny spelling	*sllooooooww, sh*t*
	URL	url, pathname or email	*http://apj.co.uk, /usr/local, phj@tpt.com*
	NONE	should be ignored	ascii art, formatting junk

SOTA AI in TTS

SOTA. Back-end.

[12]

Tacotron 2



- High quality speech
- Meets hardware constraints
- Flexible architecture

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)	$\mathbf{4.526 \pm 0.066}$

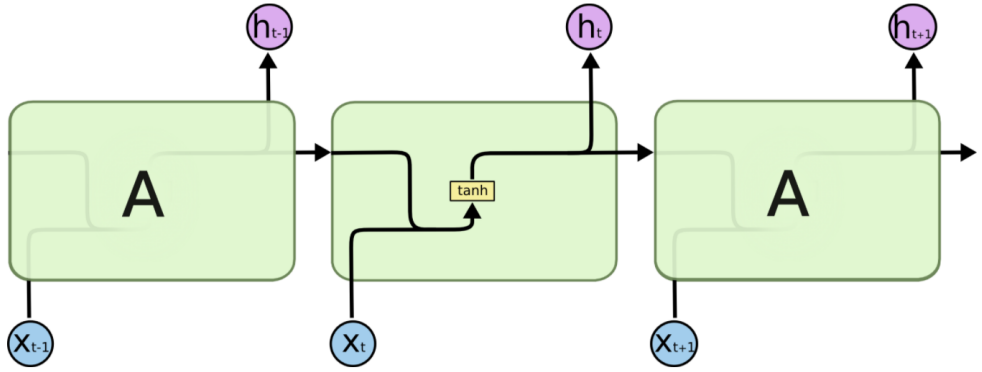
Typical TTS Server:

- CPU Intel Xeon E5-2650 v4 @ 2.20GH
- RAM - 8 Gb
- 100 parallel requests per server
- First frame delay – up to 150 ms (100*)
- Real time factor – 0.3
- Sample rate of output audio – 8000 (16000*)

* - our requiarements

SOTA. Back-end basics. LSTM.

RNN – Recurrent Neural Networks are able to train on various-length data.



The repeating module in a standard RNN contains a single layer.

- Vanishing gradients.
- Explosion of gradients.

$$c_t = \sigma\left(W_{rec} \cdot c_{t-1} + W_{in} \cdot x_t\right)$$
$$h_t = c_t$$

1. $$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial W}$$

The gradient of the error term in an RNN

$$W \leftarrow W - \alpha \frac{\partial E}{\partial W}$$

2. $$\frac{\partial E_k}{\partial W} = \frac{\partial E_k}{\partial h_k}\frac{\partial h_k}{\partial c_k} \dots \frac{\partial c_2}{\partial c_1}\frac{\partial c_1}{\partial W}$$

$$= \frac{\partial E_k}{\partial h_k}\frac{\partial h_k}{\partial c_k}\left(\prod_{t=2}^{k} \frac{\partial c_t}{\partial c_{t-1}}\right)\frac{\partial c_1}{\partial W}$$

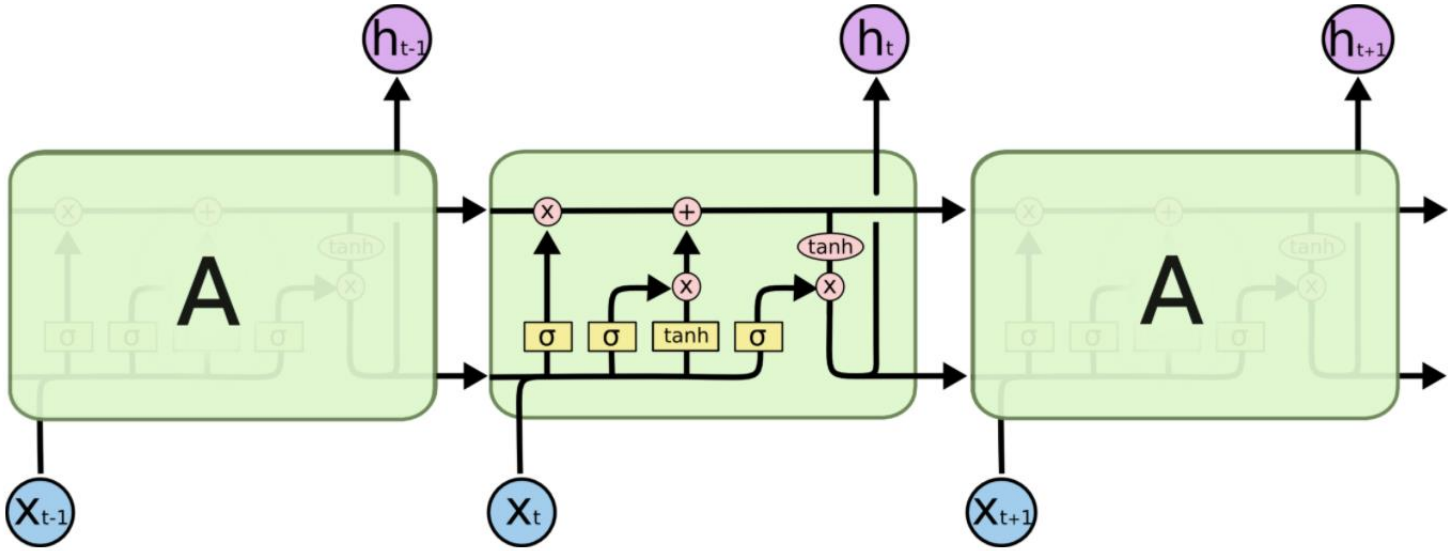
3. $$\frac{\partial C_t}{\partial C_{t-1}} = \sigma'(W_{rec} \cdot c_{t-1} + W_{in} \cdot x_t) \cdot \frac{\partial}{\partial C_{t-1}}[W_{rec} \cdot c_{t-1} + W_{in} \cdot x_t]$$

$$= \sigma'(W_{rec} \cdot c_{t-1} + W_{in} \cdot x_t) \cdot W_{rec}$$

$$\frac{\partial E_k}{\partial W} = \frac{\partial E_k}{\partial h_k}\frac{\partial h_k}{\partial c_k}\left(\prod_{t=2}^{k} \sigma'(W_{rec} \cdot c_{t-1} + W_{in} \cdot x_t) \cdot W_{rec}\right)\frac{\partial c_1}{\partial W}$$

SOTA. Back-end basics. LSTM.

RNN – Recurrent Neural Networks are able to train on various-length data.

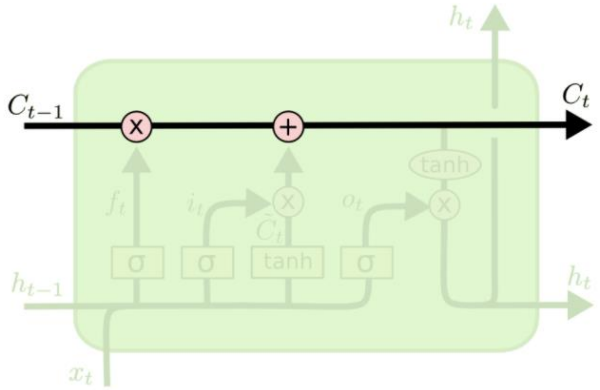


The repeating module in an LSTM contains four interacting layers.

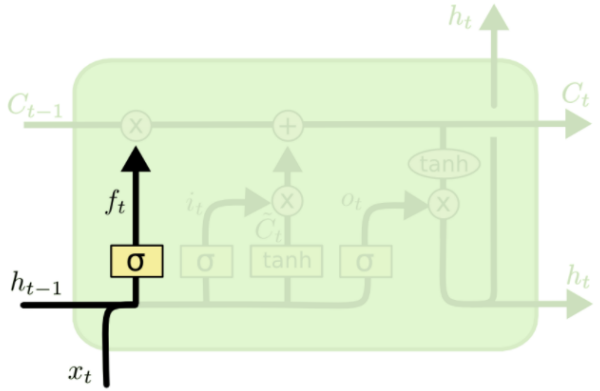
- no vanishing gradients.
- no explosion of gradients.
- increase/decrease state.
- fine control on signal propagation

SOTA. Back-end basics. LSTM.

1.

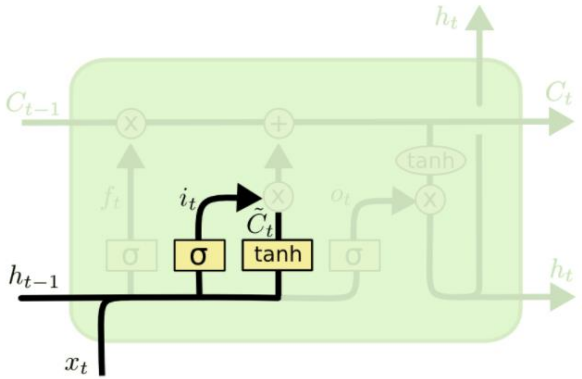


2.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

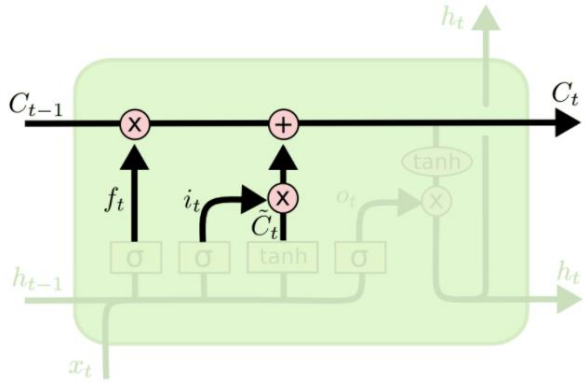
3.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

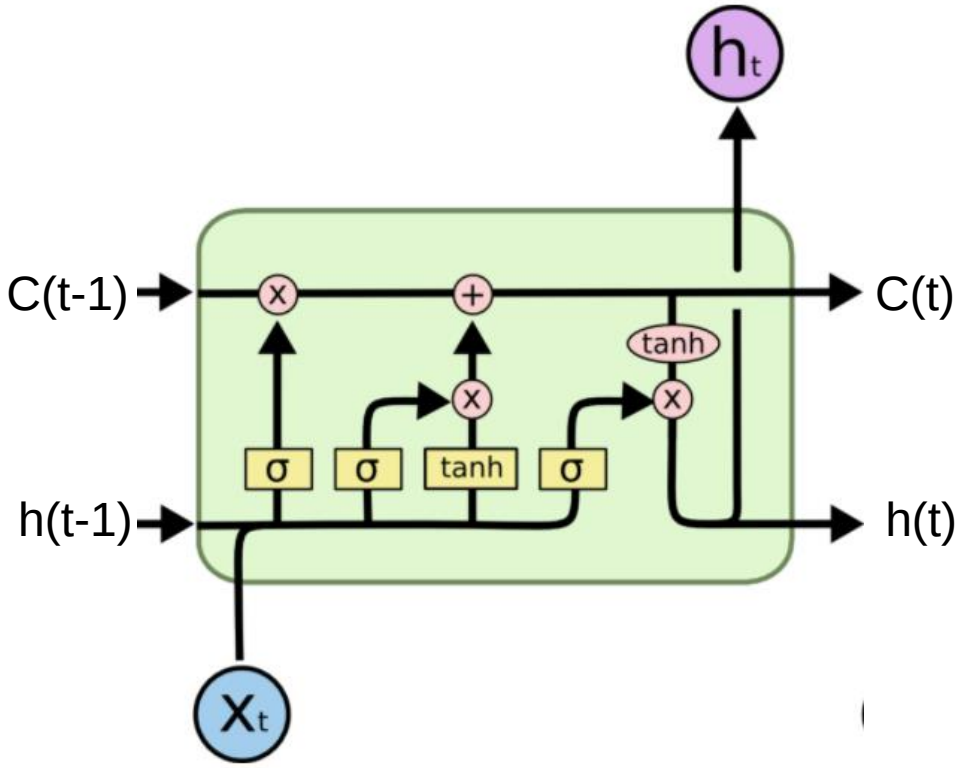
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

4.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

SOTA. Back-end basics. LSTM.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

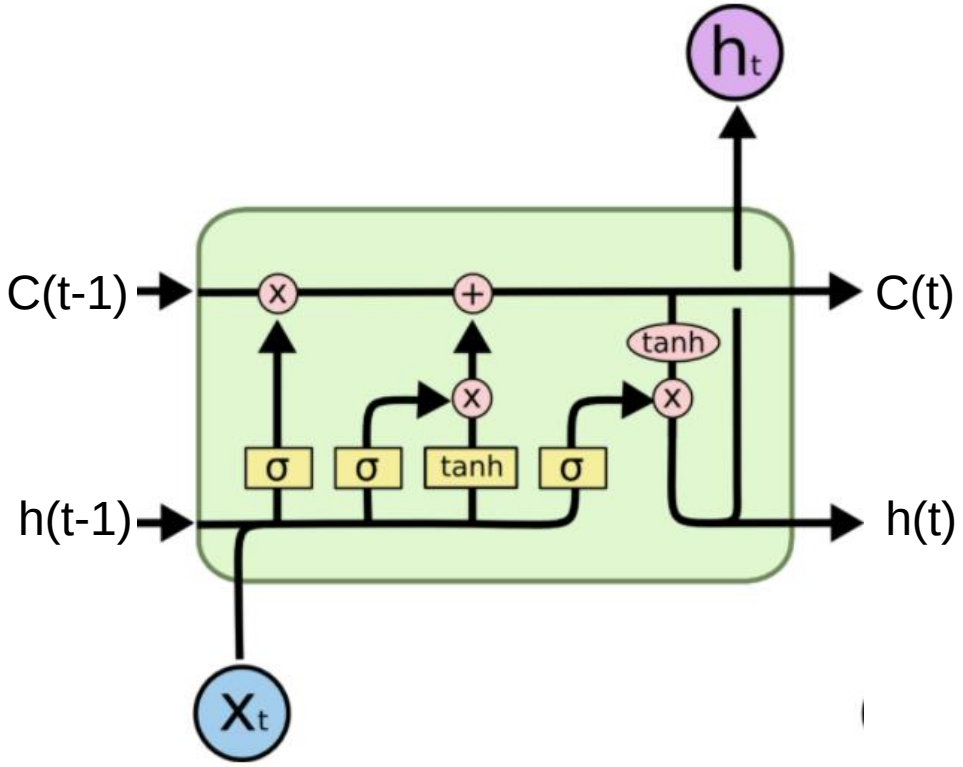
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma\left(W_o \; [h_{t-1}, x_t] \; + \; b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$

SOTA. Back-end basics. LSTM.



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$\frac{\partial c_t}{\partial c_{t-1}} = \frac{\partial}{\partial c_{t-1}}[c_{t-1} \otimes f_t \oplus \tilde{c}_t \otimes i_t]$$

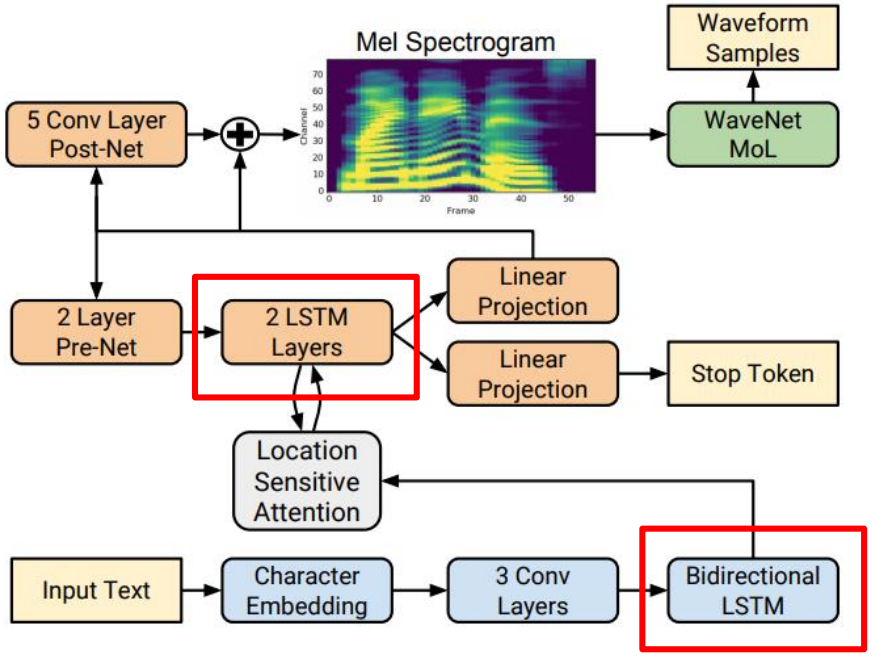
$$= \frac{\partial}{\partial c_{t-1}}[c_{t-1} \otimes f_t] + \frac{\partial}{\partial c_{t-1}}[\tilde{c}_t \otimes i_t]$$

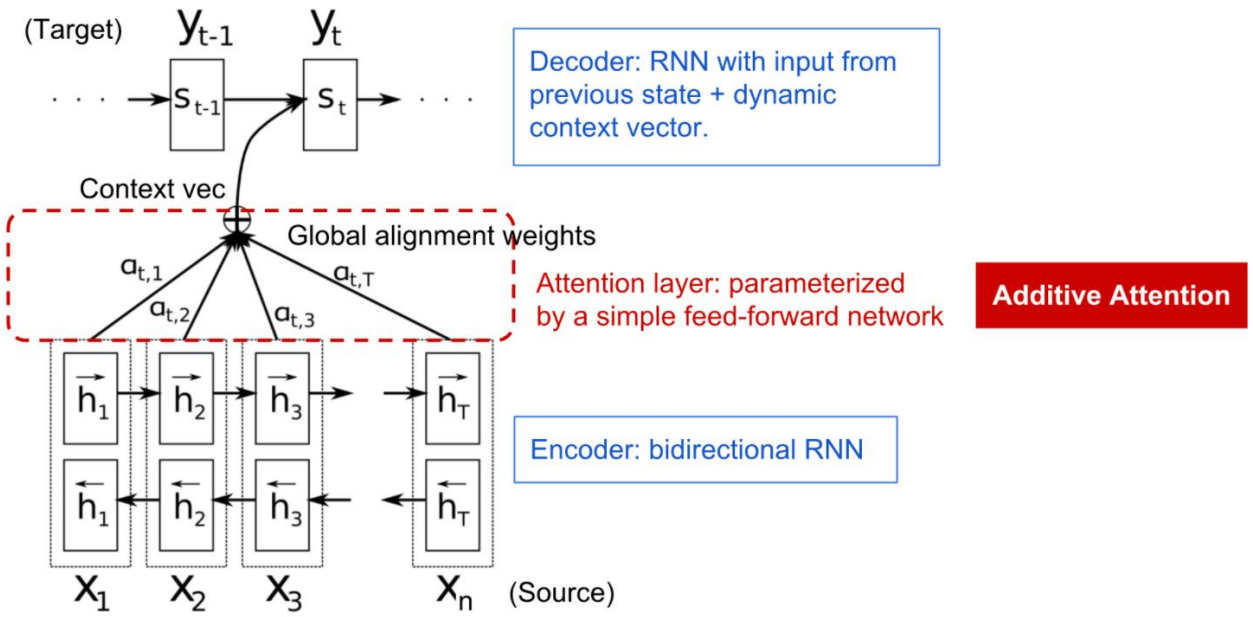
$$= \frac{\partial f_t}{\partial c_{t-1}} \cdot c_{t-1} + \frac{\partial c_{t-1}}{\partial c_{t-1}} \cdot f_t + \frac{\partial i_t}{\partial c_{t-1}} \cdot \tilde{c}_t + \frac{\partial \tilde{c}_t}{\partial c_{t-1}} \cdot i_t$$

$$\frac{\partial E_k}{\partial W} = \frac{\partial E_k}{\partial h_k}\frac{\partial h_k}{\partial c_k}\left(\prod_{t=2}^{k}[A_t + B_t + C_t + D_t]\right)\frac{\partial c_1}{\partial W}$$

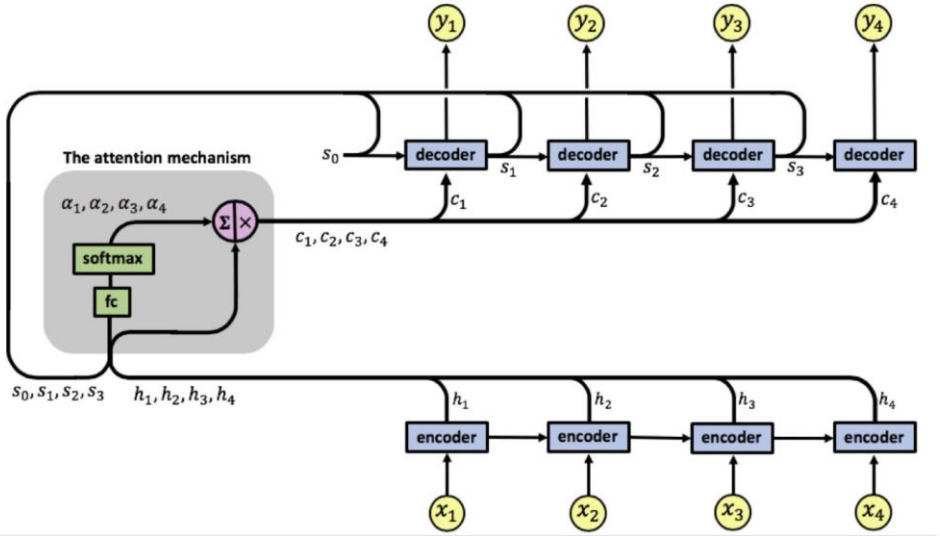
SOTA. Back-end basics. Attention.

Encoder input/output length is **n**, decoder input/output length is **m**…
Problem: **m** != **n**

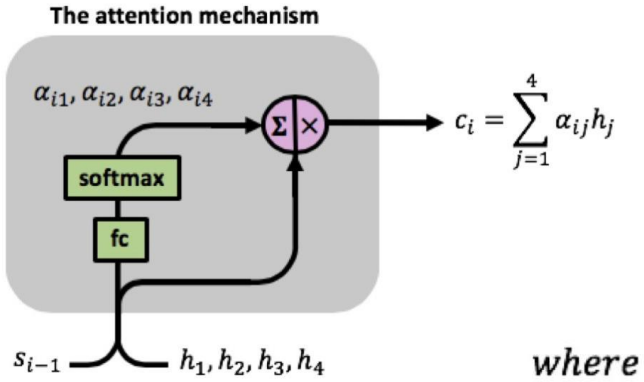




SOTA. Back-end basics.



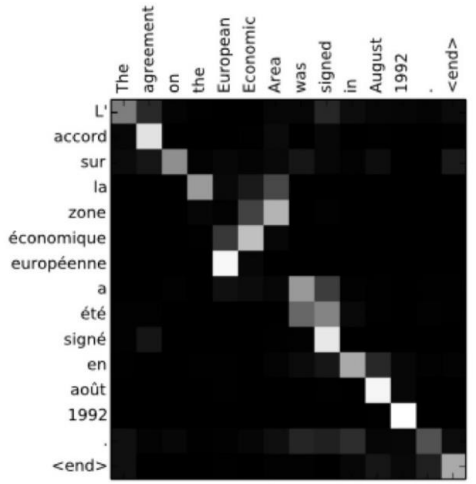
A woman is throwing a frisbee in a park

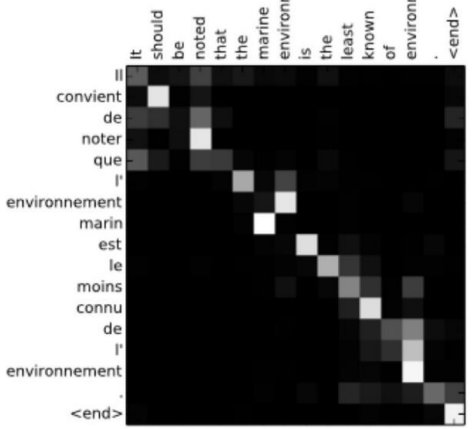


$$c_i = \sum_{j=1}^{4} \alpha_{ij} h_j$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{4} exp(e_{ik})}$$

$$where \quad e_{ij} = \mathrm{fc}(s_{i-1}, h_j)$$





SOTA. Back-end basics.

Name	Alignment score function	Citation
Content-base attention	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \text{cosine}[\boldsymbol{s}_t, \boldsymbol{h}_i]$	Graves2014
Additive(*)	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \mathbf{v}_a^\top \tanh(\mathbf{W}_a[\boldsymbol{s}_t; \boldsymbol{h}_i])$	Bahdanau2015
Location-Base	$\alpha_{t,i} = \text{softmax}(\mathbf{W}_a \boldsymbol{s}_t)$ Note: This simplifies the softmax alignment to only depend on the target position.	Luong2015
General	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \boldsymbol{s}_t^\top \mathbf{W}_a \boldsymbol{h}_i$ where $\mathbf{W}_a$ is a trainable weight matrix in the attention layer.	Luong2015
Dot-Product	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \boldsymbol{s}_t^\top \boldsymbol{h}_i$	Luong2015
Scaled Dot-Product(^)	$\text{score}(\boldsymbol{s}_t, \boldsymbol{h}_i) = \frac{\boldsymbol{s}_t^\top \boldsymbol{h}_i}{\sqrt{n}}$ Note: very similar to the dot-product attention except for a scaling factor; where n is the dimension of the source hidden state.	Vaswani2017

SOTA. Back-end basics.

Noise level control:



Level of noise can be controlled by changing one dimension, and this dimension is automatically determined by the per-dimension linear discriminative analysis

Pitch control:



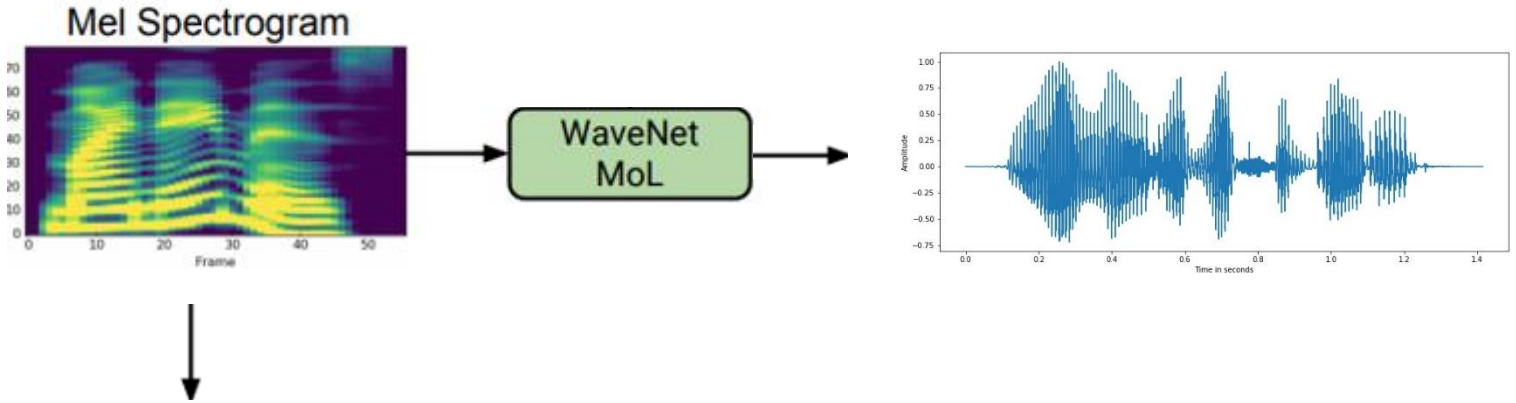
Ability of the model to synthesize speech that resembles the prosody or style of a given reference utterance.

Masculinity:



Several aspects of speaking style/prosody can be controlled by changing the value of one dimension to the latent attribute representation.

SOTA. Vocoder basics.



Amplitude spectrum does not contain phase information → Signal reconstruction is performed by vocoder

Deterministic algorithm does no show best performance

SOTA. Vocoder basics.

Gated reccurent unit:

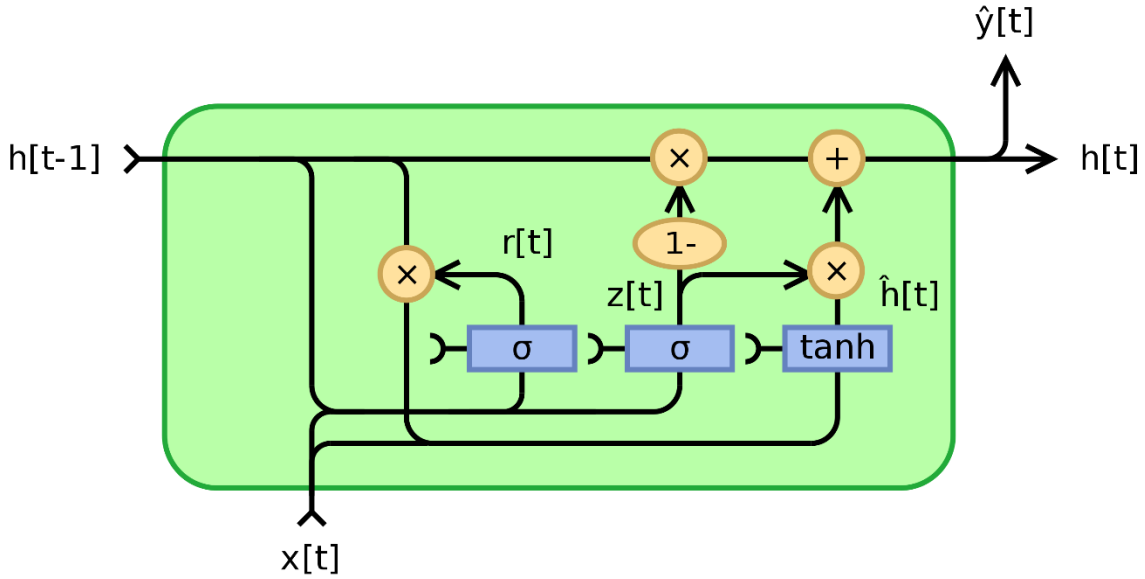
x_t: входной вектор

h_t: выходной вектор

z_t: вектор вентиля обновления

r_t: вектор вентиля сброса

W, U и b: матрицы параметров и вектор



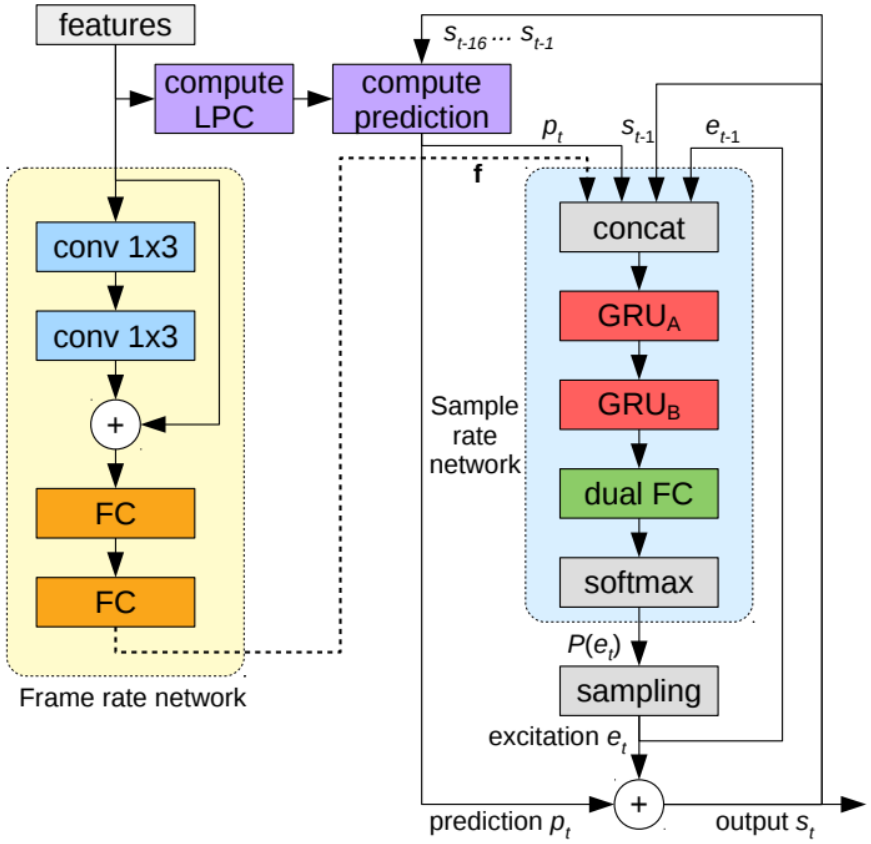
$$z_t = \sigma_g(W_z x_t + U_z h_{t-1} + b_z)$$

$$r_t = \sigma_g(W_r x_t + U_r h_{t-1} + b_r)$$

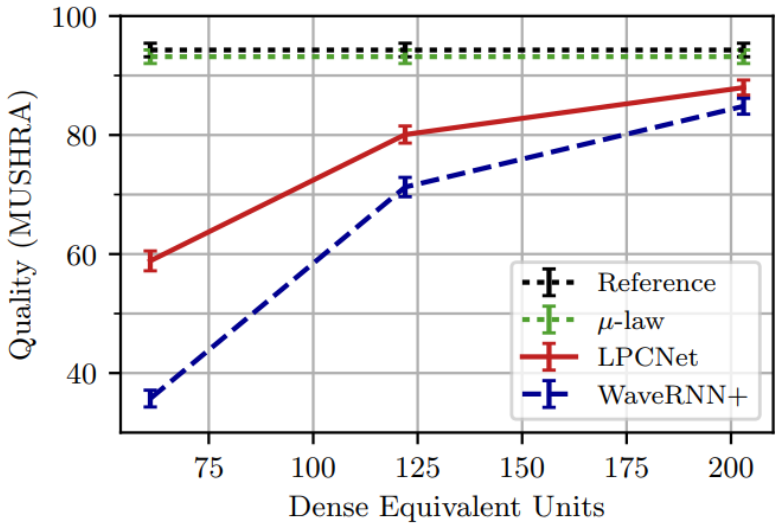
$$h_t = z_t \circ h_{t-1} + (1 - z_t) \circ \sigma_h(W_h x_t + U_h(r_t \circ h_{t-1}) + b_h)$$

SOTA. Vocoder basics.

LPCNet:



$$\mathbf{x}_t = [s_{t-1}; \mathbf{f}]$$
$$\mathbf{u}_t = \sigma\left(\mathbf{W}^{(u)}\mathbf{h}_{t-1} + \mathbf{U}^{(u)}\mathbf{x}_t\right)$$
$$\mathbf{r}_t = \sigma\left(\mathbf{W}^{(r)}\mathbf{h}_{t-1} + \mathbf{U}^{(r)}\mathbf{x}_t\right)$$
$$\widetilde{\mathbf{h}}_t = \tanh\left(\mathbf{r}_t \circ \left(\mathbf{W}^{(h)}\mathbf{h}_{t-1}\right) + \mathbf{U}^{(h)}\mathbf{x}_t\right)$$
$$\mathbf{h}_t = \mathbf{u}_t \circ \mathbf{h}_{t-1} + (1 - \mathbf{u}_t) \circ \widetilde{\mathbf{h}}_t$$
$$P(s_t) = \text{softmax}\left(\mathbf{W}_2 \,\text{relu}\left(\mathbf{W}_1 \mathbf{h}_t\right)\right) \ ,$$



SOTA. Vocoder evolution.

Vocoder	Year	Quality	Type of network	Speed	Training time
Wavenet	2016	Highest	Fully-convolutional	Extremely slow (Open versions take several minutes to produce 1 second)	Weeks
WaveRNN	2018	Highest	Fully recurrent	4x faster than realtime on GPU	Around 1 week (according to open implementations)
LPCNet	2018	High	Convolutional+Reccurent+Algorithmic(LPC)	4x faster than realtime on CPU	1-2 days
MelGAN	2019	Moderate	Fully-convolutional	20x faster than realtime on CPU	3-4 days
FeatherWave	2020	Highest	LPCNet-based	9x faster than realtime on CPU	1-2 days

Future development basis Disentaglement and control.

Disentaglement

Problem

Speech data contains a lot of information:

$$\text{Speech data} = \text{Text} + \text{Speaker} + \text{Prosody} + \text{Recording conditions}$$

some of theme are present in the annotation and some are not.

During training model learns to model ALL components of the speech relying only on the annotation. That results in:

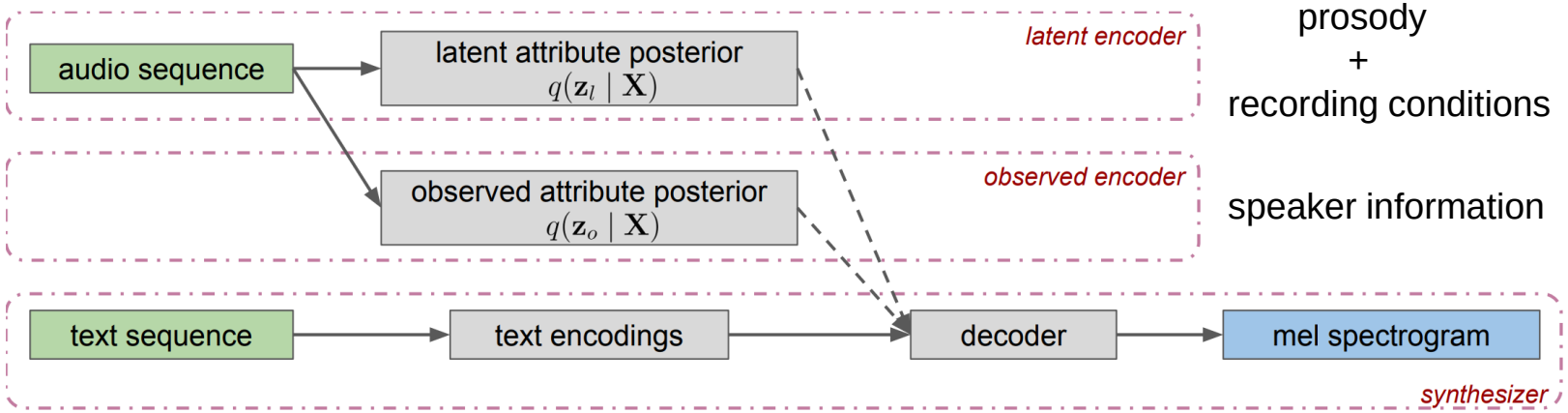
- random or flat prosody,
- some level of noise during synthesis if recording artifacts were present in the data.

1. We want to control all components of the speech during synthesis.
2. We want to train TTS without additional mark-up.

Disentaglement

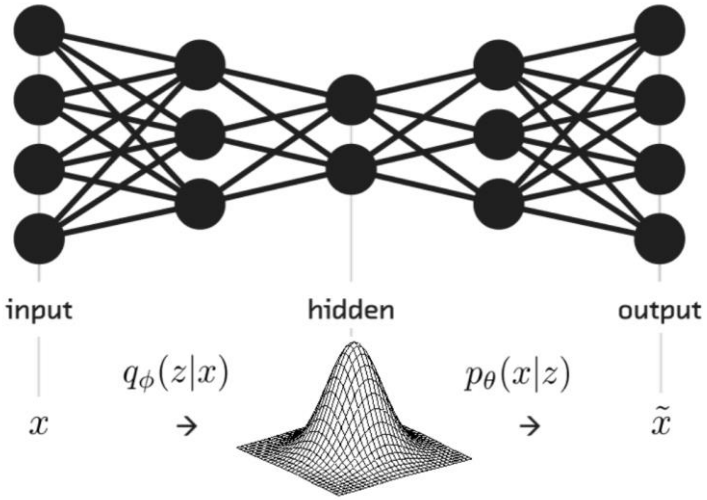
[11]

Towards automatic semantics capture:



prosody
+
recording conditions

speaker information



When evaluating the meaning of each dimensions, authors found the majority of the dimensions to be interpretable, and the number of dummy dimensions which do not affect the model output varied across datasets

GMVAE:

- uses hierarchical model for latent variables
- most of the component has interpretable role (noise/speed/pitch/accent etc) – easy control

VAE:

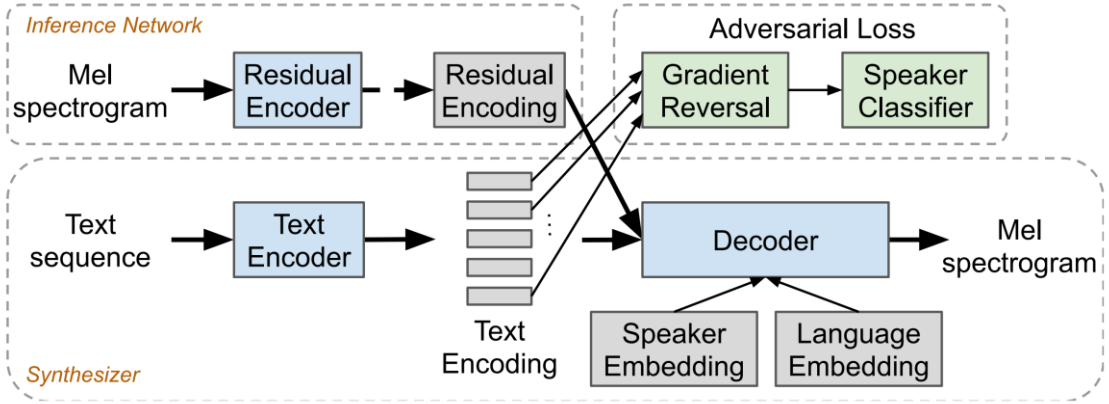
- Latent space is not interpretable:
  Separate dimensions of the vector have no meaning
- Due to the above control (in practice) is problematic

Multi-lunvo multi-speaker model

Current project results:

Multi-lingvo multi-speaker model PoC:

http://10.199.125.117:9999/notebooks/notebooks/200703_multilanguage-denis.ipynb



Target:

MSML: 10 speakers per each of 10 languages
Size: <50% bigger that single speaker model
Quality: same MOS as single speaker model

Challenges:

Speaker/accent/language disentanglement
NO product-ready open technology

Key technology:

VAE + RNN combination

original
Spanish
French
Russian

Future development basis. Disentaglement impact to joining tasks.

Impact on joining domains: disenteglement in CV

A Disentangling Invertible Interpretation Network for explaining Latent Representations **[1]**.

Usage invertible translation to extract semantics.

No restrictions on network f under-the-hood.

$$f(x) = G \circ E(x)$$

$$training\ pair\ (x_a, x_b)\ represents$$

$$semantic\ factor\ F \in \{0, \dots, K\}$$

$$\check{z} \rightarrow \check{z}_1, \dots, \check{z}_K$$

$$p(\check{z}) = \prod_0^K N(\check{z}_i|0, I)$$

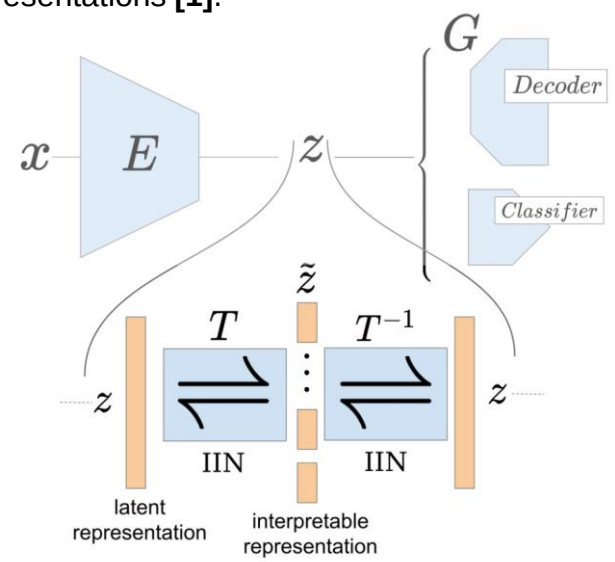
$$1\ Rule:\ \check{z}_i\ varies\ only\ 1\ semantic\ concept$$

$$\check{z}_F^b \sim N\left(\ \check{z}_F^b\ \middle|\ \sigma_{ab} * \check{z}_F^a, (1 - \sigma_{ab}) * I\ \right)$$

$$2\ Rule:\ \check{z}_i\ invariant\ to\ other\ variations$$

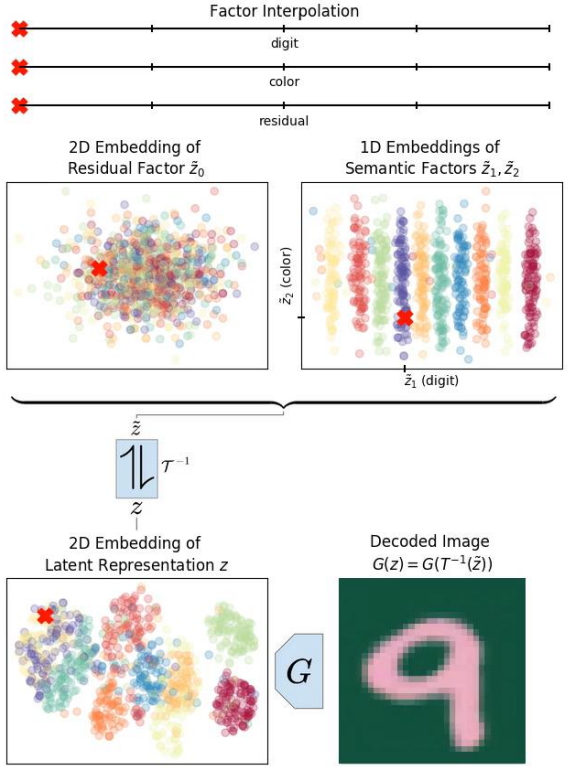
$$\check{z}_F^b \sim N\left(\ \check{z}_k^b\ \middle|\ 0, I\ \right), for\ \ k \in \{0, \dots, K\} \backslash \{F\}$$

1.

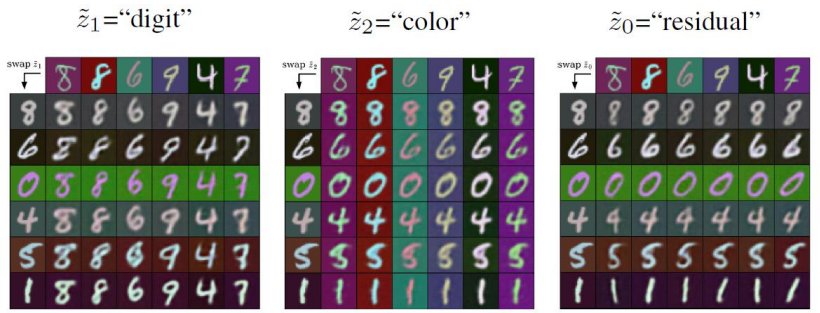


Interpretable representation $\tilde{z}$ without information loss.

3.



2.



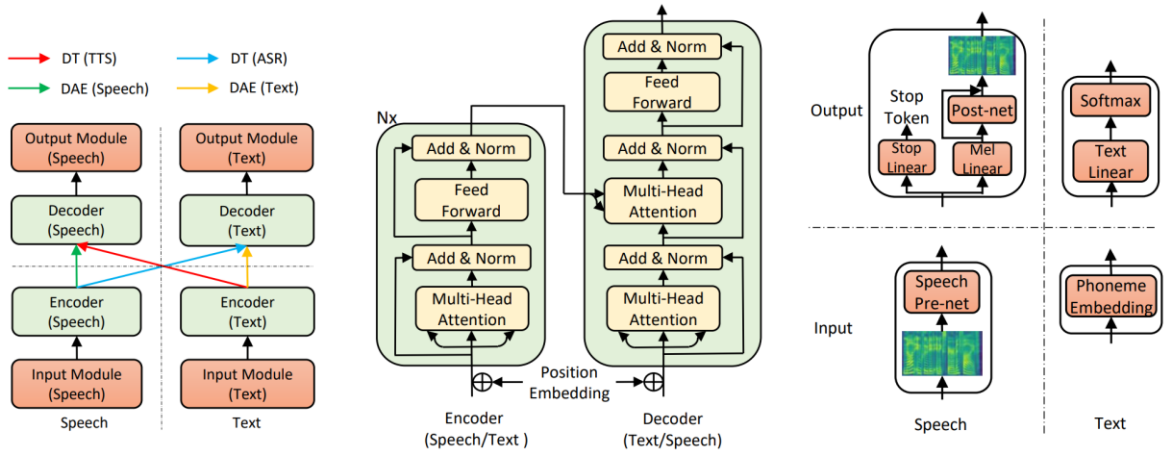
Interpolate within individual semantic concepts and visualize representations embedded onto semantically-meaningful dimensions.

Future development basis
Improve ASR.

TTS improves speech recognition

Almost Unsupervised Text to Speech and Automatic Speech Recognition **[2]**

Authors used mutual training of unified transformers architecture for TTS, ASR.



(a) Unified training flow (b) Encoder/Decoder for speech/text (c) Input/Output module for speech/text

Figure 1. The overall model structure for TTS and ASR. Figure (a): The unified training flow of our method, which consists of a denoising auto-encoder (DAE) of speech and text, and dual transformation (DT) of TTS and ASR, both with bidirectional sequence modeling. Figure (b): The speech and text encoder and decoder based on Transformer. Figure (c): The input and output module for speech and text.

Method	MOS (TTS)	PER (ASR)
GT	4.54	-
GT (Griffin-Lim)	3.21	-
Supervised	3.04	2.5%
Pair 200	Null	72.3%
Our Method	2.68	11.7%

Table 1. The comparison between our method and other systems on the performance of TTS and ASR.

Paired Data	100	200	300	400	500
PER (ASR)	64.2%	11.7%	8.4%	5.2%	4.4%
MOS (TTS)	Null	2.45	2.49	2.64	2.78

Table 3. The PER on ASR with different amount of paired data for our method.

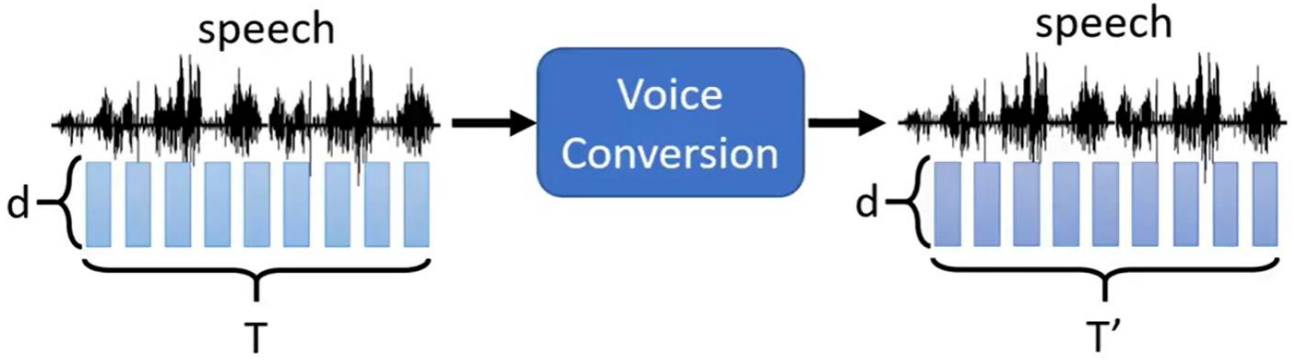
Authors achieved 99.84% in terms of word level intelligible rate and 2.68 MOS for TTS, and **4.4% PER for ASR with just 500 paired data on LibriSpeech** dataset. They stated to prove that it is possible to train ASR with close to SOTA performance only with 500 audio clips (~1 hour).

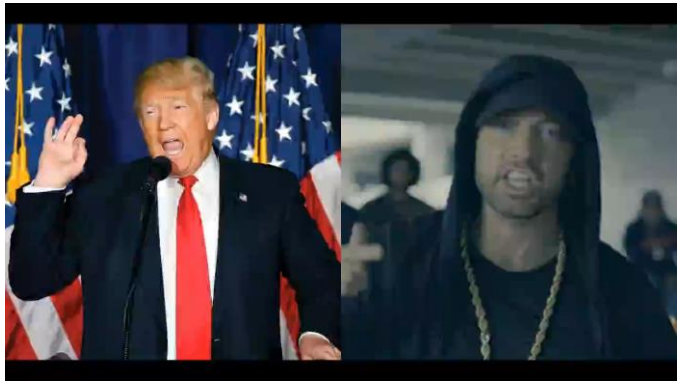
Future development basis. Conversion and cloning.

Voice conversion

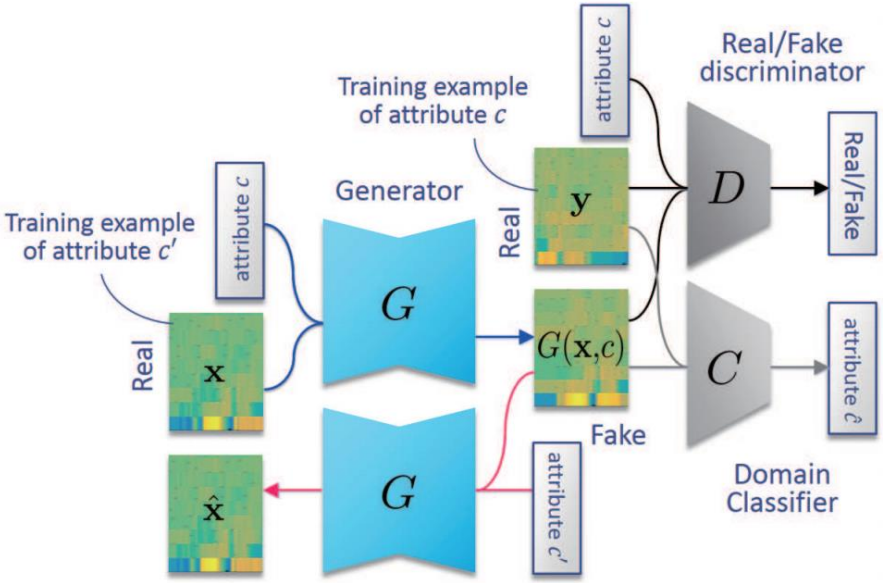
- Audio to audio task
- Given an audio, change the voice of the speaker to target voice
- All other parameters (text pronounced, prosody, sound conditions) should stay the same
- New voice sample should be provided

- Popular approach is to use
- GAN architechture (StarGAN)
- as in **StarGAN-VC:**
- **Non-parallel many-to-many voice conversion with star generative adversarial networks**





Donald Trump AI Model Raps 'Lose Yourself' by Eminem



Concept of StarGAN training.

Voice cloning

[10]

- TTS System, where target speaker voice sample is very small
- Voice sample can vary from several seconds to tens of minutes
- Few shot (Single shot) voice cloning is a system that does not require additional training after getting a voice sample (Current systems have very low voice similarity in this case)
- Popular approach is to use a separate encoder networks, which computes an embedding for the speaker based on small voice-sample, and then using the embedding in tacotron

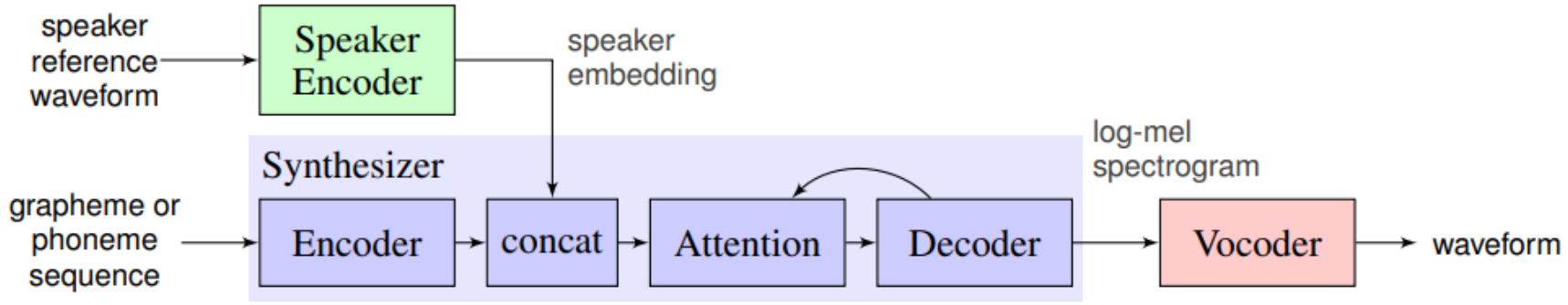


Figure 1: Model overview. Each of the three components are trained independently.

Future development basis Emotional TTS.

Emotional TTS

Audio-based	Computer vision-based	Behavioral-based/-mixed
audEERING™ intelligent Audio Engineering	emteq	IMOTIONS BIOMETRIC RESEARCH PLATFORM
Receptiviti	Robbie AI	emaww EMOTIONS MATTER
yobe	eyeris	SENSUM
pindrop	Pyramics	
cogito	:) Affectiva	
Intelligent Voice	EMOSHAPE EMOTIONS NEXT FRONTIER	
:) Affectiva	soul machines	
VoiceSense A window to your mind	Microsoft	
nemesysco voice analysis technologies		

Proprietary Technology

amazon · Microsoft · facebook · Google · Apple

Some vendors use multiple data input sources to train their systems for emotion AI recognition

Emotional TTS

CBG TTS team cooperation projects:

1) *Speech Emotion Recognition (2020-2021)*
   - *by voice*
   - *by text*
2) *Emotional Text-to-Speech System (2020-2022)*

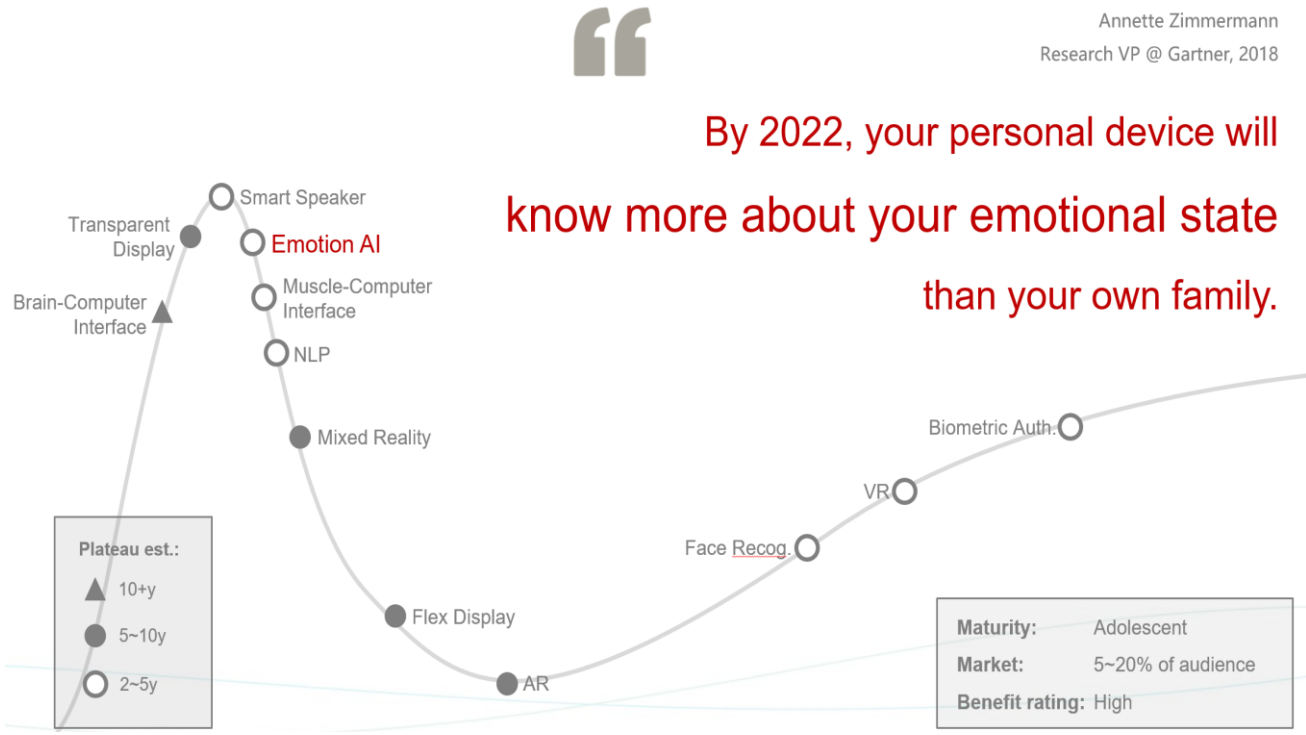
Challenges:

poorly defined task
- not determined basic emotion list

dataset transfer problems
- language transfer
- audio/text/video transfer
- actors/spontaneous speech transfer
- different labeling process

model robustness and stability

NO product-ready open technology

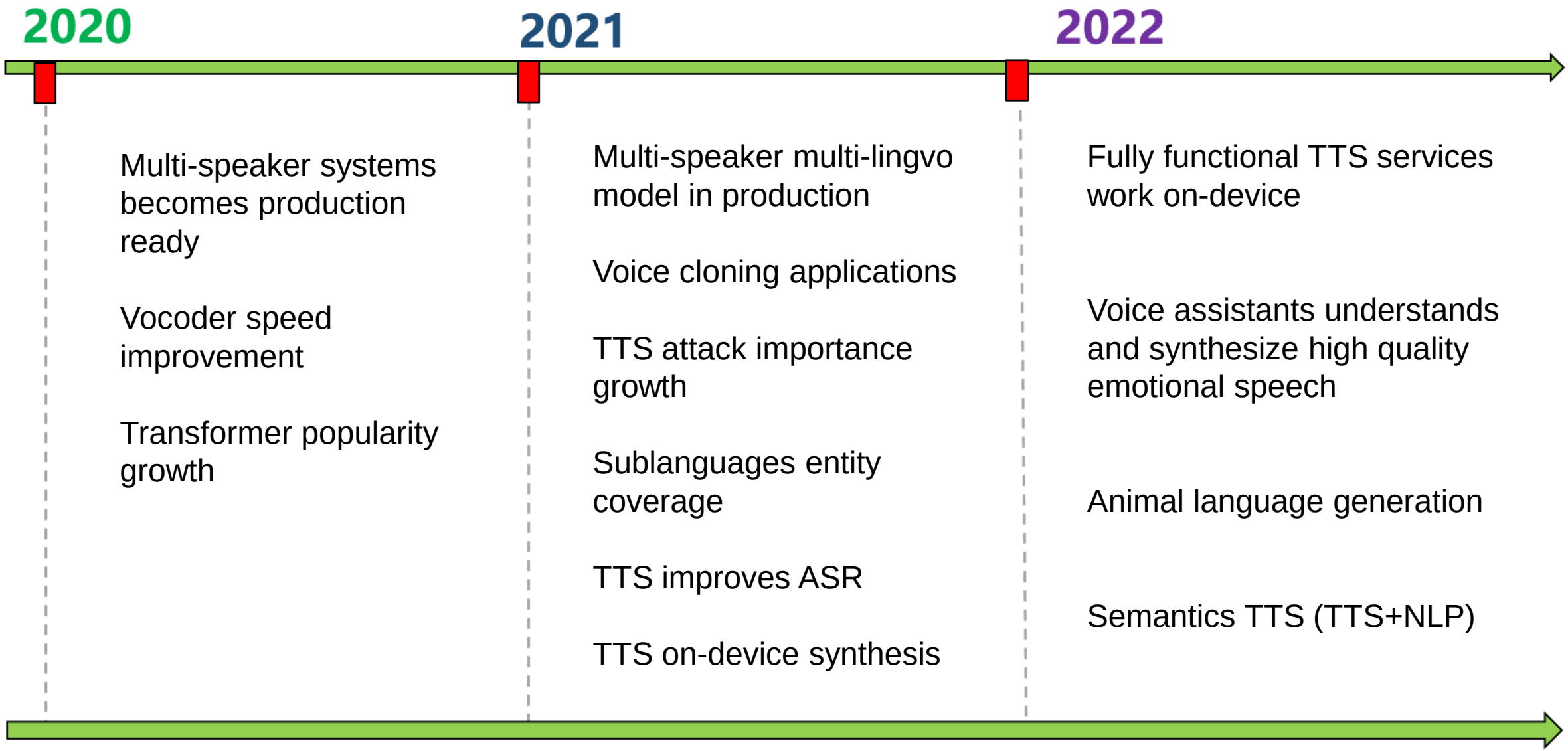


Key technology:

pretrained features + LSTM/CNN combination

Technological Forecast.

Global trends



Thank you!