Tutorial on Speech-to-Speech Machine Translation

ICON2021

Presenters: Akshay Batheja, Rohit Kundu, Vineet Bhat, Shivam Mhaskar, Shyam Thombre, Sourabh Deoghare, Tamali Banerjee, Jyotsana Khatri,

and

Prof. Pushpak Bhattacharyya

IIT Bombay

Date: 19/12/2021

Contents

- Speech-to-Speech Machine Translation (SSMT)
- Automatic Speech Recognition (ASR)
- Disfluency Correction (DC)
- Machine Translation (MT)
- Automatic Post Editing (APE)
- Text-to-Speech (TTS)
- SSMT Demo

Speech-to-Speech Machine Translation

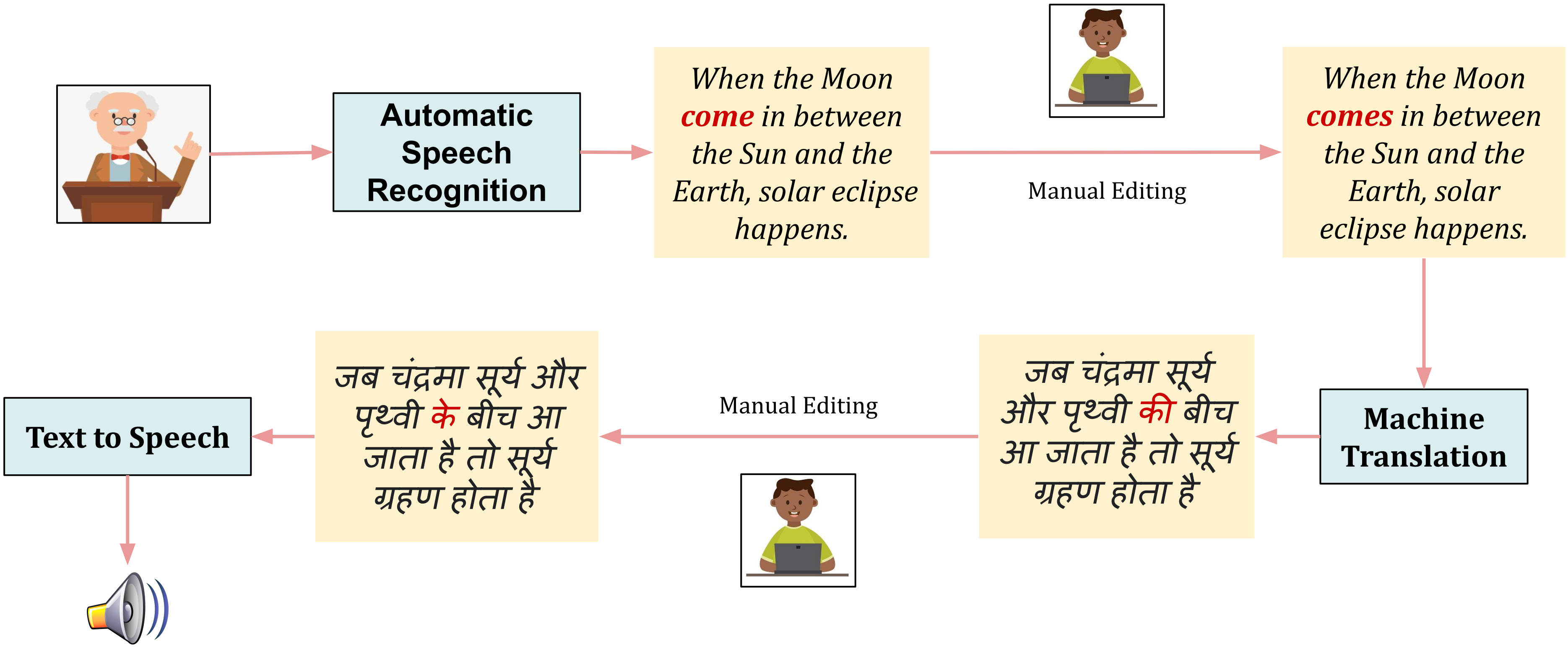
Problem Statement

- Speech-to-Speech Machine Translation : To translate speech in language A into speech in language B through use of a computer.

Speech segment in L1

Speech segment in L2

NLTM: Bahubhashak Project: Speech to Speech Machine Translation - Pipeline



Motivation

- SSMT technologies witnessing rapid growth:
  - Globalization, business needs, frequent travel, tourism industry fueling its demands.

- Huge range of applications, examples:
  - Movie dubbing
  - Movie Subtitling
  - Conversing with foreign language speakers

SSMT Market: USD 330 million in 2020 → USD 600 million in 2026

Study Period:	2018 - 2026
Base Year:	2020
Fastest Growing Market:	Asia Pacific
Largest Market:	Europe
CAGR:	9.4 %

https://www.mordorintelligence.com/industry-reports/speech-to-speech-translation#:~:text=The%20speech%20to%20speech%20translation%20market%20is%20valued%20at%20USD,period%20(2021%2D2026).

Benefits

- Bridging the language gap in global commerce and cross-cultural communication
  - Aids communication between people speaking different languages
- Education in mother tongue
  - Will dramatically reduce time for making Lectures available in Indian Languages
- Skills development amongst youth
- Technology could be scaled to school education
  - A promising solution for imparting quality school education in rural India

Challenges

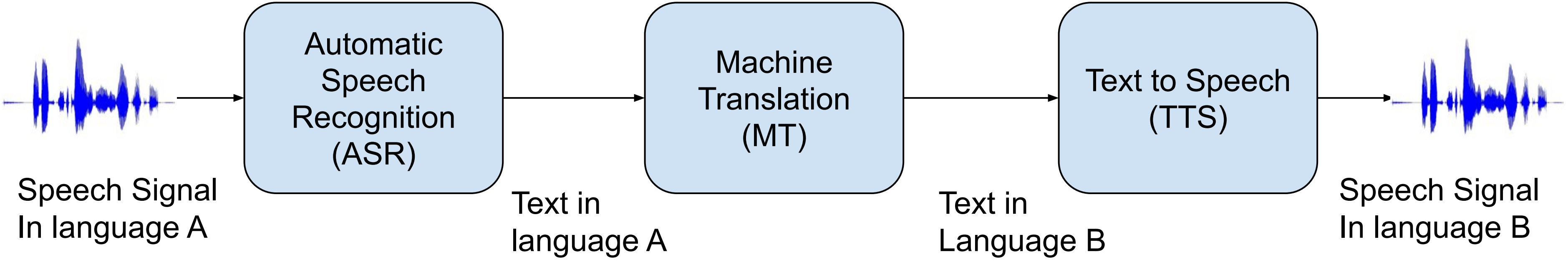
- Conversational Spontaneous Speech
  - *It is a ahh umm beautiful day*
  - Ill-formed and incomplete sentences
- Indian English
  - Accent varies across the Nation
  - A problem for ASR
- Word Ordering
  - Significantly different for English and Indian languages
  - A problem for MT
- Code Switching
  - *आज I am busy*
- Lip Syncing
  - Translated sentences to Indian Languages are usually longer.

Cascaded SSMT Approach

ASR : Transcribing speech into text

MT : Translating text from language A to language B

TTS : Synthesizing speech from text



Well known Systems

US: Janus, DIPLOMAT, Tongues, Nespole! Maxtor

Europe: Verbmobil, Nespole! LC-Star, TC-Star

Japan: MATRIX NEC

China: LodeStar, Digital Olympics

SSMT: a Data Driven S2S problem

- Training data for an end-to-end ST model is very scarce.
- Available currently: only hundreds of hours of speeches, most of which are for
- Japanese–English translation and European languages [102,103]
- For Chinese–English translation, Baidu has released an open dataset containing 70 hours of speeches, including both the corresponding transcriptions and translations
- Through MEITY funded speeech consortia led by IIT Madras, speech data in Indian lanaguges is now available

Direct Speech to Speech Machine Translation

Direct Speech to Speech MT

- Problem Statement
  - To translate speech input in one language into speech in another language without relying on the intermediate step of text generation.

- Motivation
  - Lower computational costs and inference latency as compared to the cascaded systems.

  - To provide a translation support for languages that do not have a writing system.

Previous Approaches

- Cascaded S2S MT (2006) [2]
  - The Cascaded S2S pipeline consists of three components
    - ASR -> MT -> TTS

- End-to-End S2T (Speech to Text) (2019) [3]
  - Alleviates the error propagation issue between ASR and MT

  - S2T can be combined with TTS to provide both Speech and Text translation

Recent Developments

- Translatotron by Google AI (2019)
  - An attention-based sequence-to-sequence neural network which can directly translate speech from one language into speech in another language
- Direct S2ST with Discrete Units by Facebook AI and Johns Hopkins University (2021) , state-of-the-art in Direct S2ST

Direct S2ST with Discrete Units (1/2)

- Key Contributions:
  - Self-supervised discrete representations of target speech is predicted instead of mel spectrogram features.
  - Model jointly generates speech and text output by combining S2ST and S2T tasks through the shared encoder and a partially shared decoder, for the languages where the transcripts are available at source as well as at target.

Direct S2ST with Discrete units (2/2)

- Key Contributions:
  - In the scenarios where the transcripts for target language is unavailable, Direct S2ST model is trained with multitask learning using discrete representations for the source and target speech.
  - The issue of length mismatch between the text and speech output during decoding is resolved using CTC (connectionist temporal classification).

Model Architecture (1/2)

- Transformer-based seq-to-seq model with a speech encoder and a discrete unit decoder and incorporates auxiliary tasks similar to *translatotron* during training to facilitate model learning.

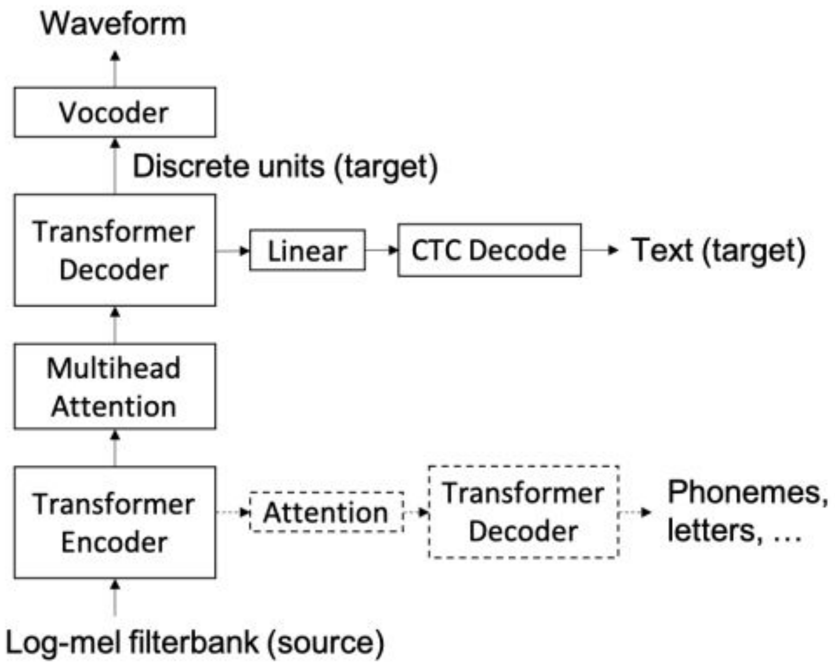


Image source -Discrete Speech to Speech Translation [2]

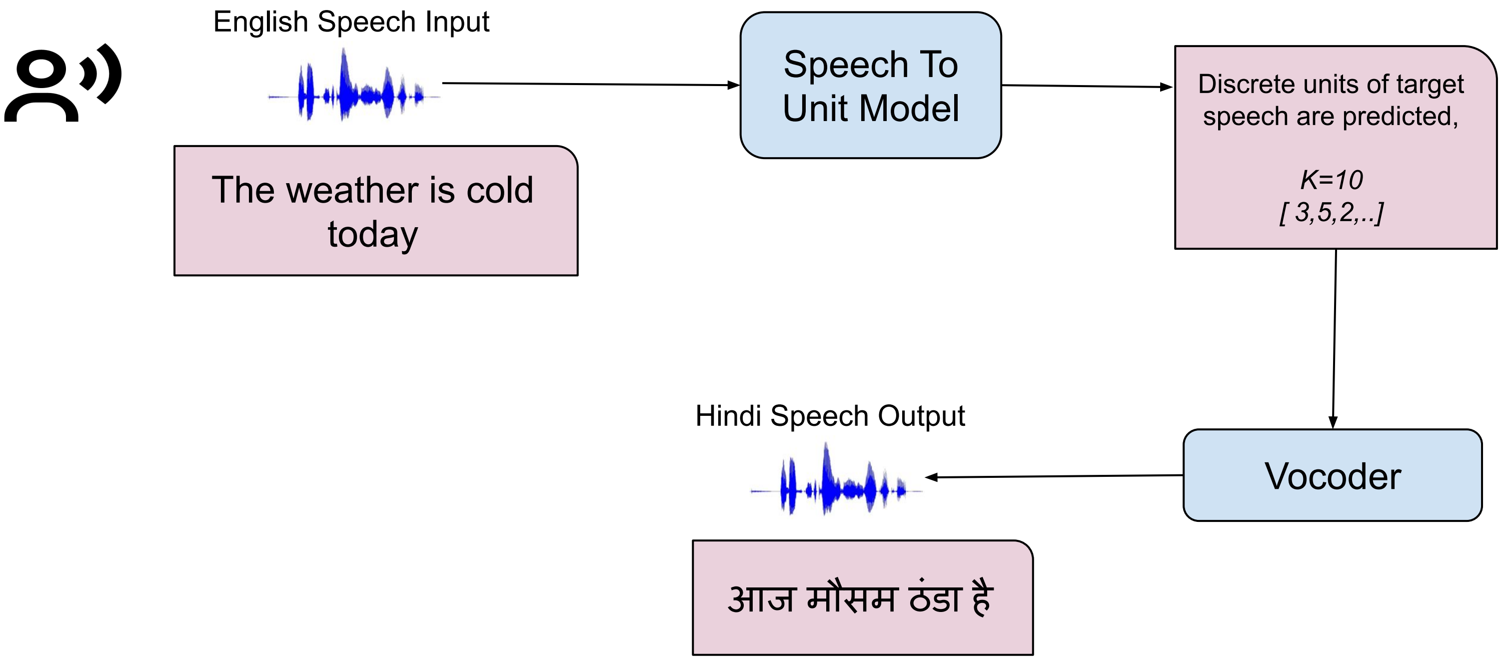
Model Architecture (2/2)

- For target languages with text transcripts, target text CTC decoding is applied conditioned on the intermediate representations from the discrete unit decoder.

- This CTC decoding is used for jointly training text and speech.

- A Vocoder is separately trained to convert discrete units into a waveform.

Direct S2ST Pipeline



Speech to Unit (S2U) Model (1/2)

- Generation of Discrete Units for the target language:
  - HuBERT model is trained on an unlabelled speech corpus of the target language.
  - The trained model is used to encode the target speech into continuous representations at every 20ms frame.
  - A k-means algorithm is applied on the learned representations to generate K cluster centroids.
  - These K cluster centroids are used to encode the target utterances into sequences of cluster indices at every 20 ms.

Speech to Unit (S2U) Model (2/2)

- A target utterance y is represented as $[z_1, z_2, .., z_T]$, where z_i belongs to $\{0, 1, 2, .., K\text{-}1\}$, K is the number of clusters and T is the number of frames.
- S2U model is built by adapting from the transformer model for MT (Machine Translation).
- A stack of 1-D convolutional layers, each with stride 2 and followed by a gated linear unit activation function, is prepended to the transformer layers in the encoder for downsampling the speech input.
- As the target sequence is discrete, S2U model is trained with cross entropy loss with label smoothing.
- Trained the Hi-Fi GAN neural vocoder for unit-to-waveform conversion.

Dataset used (1/2)

- Fisher Spanish-English speech translation corpus

  - The dataset consists of 139k sentences from telephone conversations in Spanish, the corresponding Spanish text transcriptions and their English text translation.

- Data Preparation:
  - A high-quality TTS engine is used to prepare synthetic target speech with a single female voice as the training targets.

Dataset used (2/2)

- Train, dev, test split:

	train	dev	dev2	test
# samples	126k	4k	4k	3.6k
source duration (hrs)	162.5	4.6	4.7	4.5
target duration (hrs)	139.3	4.0	3.8	3.9

Results (1/3)

- Source and Target text transcripts are available:
  - For the systems with dual mode output (like cascaded and S2U + CTC), both, the text output directly from the system and the ASR decoded text from the speech output, are evaluated.

Corpus	Models	Test (BLEU)	
		Speech	Text
Fisher Spanish-English Speech Translation Corpus	Translatotron + Pretrained encoder	31.1	-
	Cascaded (S2T + TTS)	39.5	41.5
	S2U reduced + CTC (w/ sc,tc)	**37.2**	**39.4**

Results (2/3)

- Source text transcripts available but target text transcripts are unavailable:
  - All models are trained without using any target text transcripts.

Corpus	Models	Test (BLEU)
Fisher Spanish-English Speech Translation Corpus	Translatotron (w/ sp)	7.2
	S2U reduced (w/ sc)	**33.8**

Results (3/3)

- Both source and target text transcripts are unavailable:
  - All models are trained without using any text transcripts.

Corpus	Models	Test (BLEU)
Fisher Spanish-English Speech Translation Corpus	Translatotron no auxiliary task	0.6
	S2U reduced (w/ su)	**27.1**

Summary

- In scenario where text transcripts are available at source as well as target, S2U reduced model with joint speech and text training and auxiliary tasks, has bridged 83% of the gap between transformer-based Translatotron and the S2T+TTS cascaded baseline.

- Also demonstrated the possibility of translating between two unwritten languages by taking advantage of discrete representations of both the source and the target speech for model training.

Comparing Direct S2ST with Cascaded approach

Advantages	Disadvantages
Cascaded system have problem of errors compounding between components e.g. Recognition errors leading to larger translational errors. Direct S2ST model does not face such issues.	A large set of Input/Output speech pairs are required which are more difficult to collect than parallel text pairs for MT.
Reduced computational requirements and lower inference latency.	Cascaded S2ST is more robust.
Paralinguistic and Non-linguistic information is retained during translation.	Uncertain alignment between two spectrograms whose underlying spoken content differs also poses a major training challenge.

References

[1] Ye Jia and Ron J. Weiss and Fadi Biadsy and Wolfgang Macherey and Melvin Johnson and Zhifeng Chen and Yonghui Wu. 2019. Direct speech-to-speech translation with a sequence-to-sequence model. In Interspeech.

[2] Ann Lee and Peng-Jen Chen and Changhan Wang and Jiatao Gu and Xutai Ma and Adam Polyak and Yossi Adi and Qing He and Yun Tang and Juan Pino and Wei-Ning Hsu. 2021. Direct speech-to-speech translation with discrete units. In Interspeech.

[3] Mattia Antonino Di Gangi, Matteo Negri, Marco Turchi. 2019. One-to-many multilingual end-to-end speech translation. IEEE

[4] Satoshi Nakamura, Konstantin Markov, Hiromi Nakaiwa, Genichiro Kikui, Hisashi Kawai, Takatoshi Jitsuhiro, Jin-Song Zhang, Hirofumi Yamamoto, Eiichiro Sumita, and Seiichi Yamamoto. The ATR multilingual speech-to-speech translation system. 2006. IEEE

Automatic Speech Recognition

Topics to be covered in ASR

- A general introduction to ASR: Problem statement to mathematical description
- Discussion regarding the Hidden Markov Model based approach to ASR
- Introduction to Deep Learning based ASR
- Techniques for developing good quality speech recognition systems for Indian languages
- Code walkthrough for a SOTA training and evaluation pipeline for English ASR + Demo for Hindi ASR

Automatic Speech Recognition

- ASR is the task of using algorithms and methodologies to enable translation of speech signals to text by computers.
- Research has developed from 1960s to 2020s
- Speech recognition technologies have wide scale use in education, software development, utilities, luxury, military, etc.
- Well known examples: YouTube closed captioning, Voicemail transcription, Dictation Systems, etc

Simplified pipeline of an ASR System

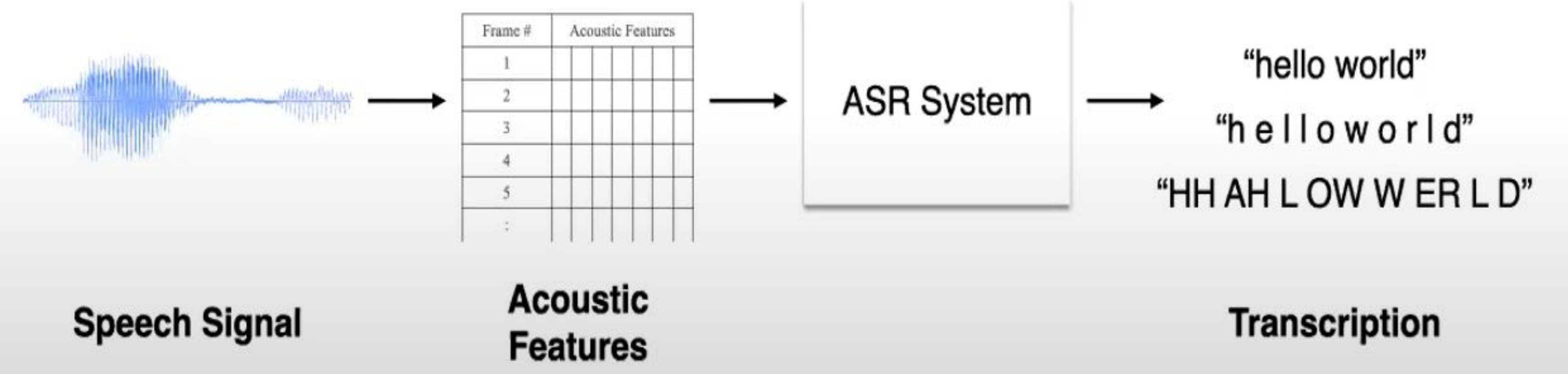
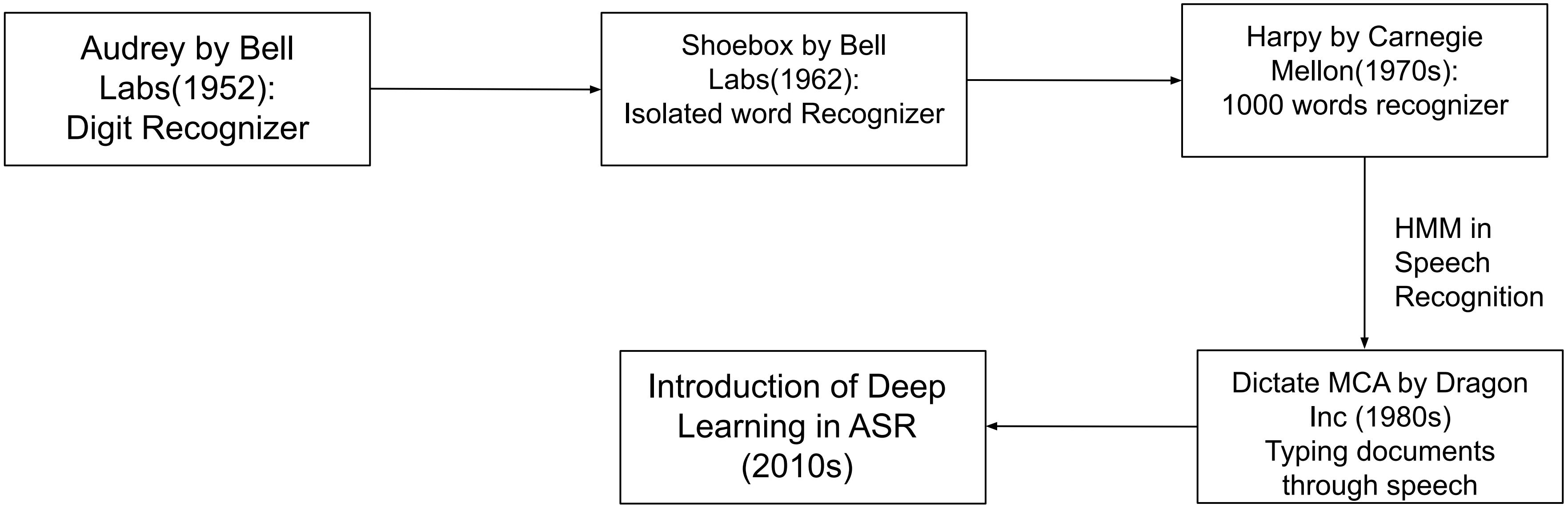


Image source: Lectures on Automatic Speech Recognition by Prof Preethi Jyothi, Course CS753, IIT Bombay

Motivation

- ASR Research has gained momentum over the years with the advent of faster computation and better resources
- Extensive data collection across widely spoken languages
- Disparity in Data availability for regional languages
- Innovative methods such as Transfer Learning and Knowledge sharing yet to be explored fully

Prior Work (Pre-Deep Learning era)



Mathematical Description of ASR

- We treat the acoustic input signal as $X = \{x_1, x_2, x_3, x_4, \ldots\}$ a series of observations and define a sequence of words as the desired output $W = \{w_1, w_2, w_3, w_4, \ldots\}$
- Essentially, we are interested in obtaining the following -

$$\boxed{\hat{W} = \underset{W \in L}{\arg\max}\, P(W|X)}$$

- We can use Bayes rule to rewrite this as -

$$\hat{W} = \arg\max_{W \in L} \frac{P(X|W)P(W)}{P(X)} \longrightarrow \hat{W} = \arg\max_{W \in L} P(X|W)P(W)$$

- P(X|W) : Probability of occurrences X given words W (Acoustic Models)
- P(W): Probability associated with word sequence (Language Models)

Acoustic Models:

- Contains statistical representations of each of the distinct sounds that make up a word
- 44 phonemes in English, each phoneme has its own HMM

Lexicon Models:

- Pronunciation models for speech recognizers; provides discriminating metric between pronunciations of the same word in different context
- Eg, consider “ough” as in through, dough, cough, rough, bough, thorough, enough, etc

Language Models:

- Predicting next word given a sequence of words
- Used to calculate P(W) in previous expressions

Architecture of cascaded ASR System

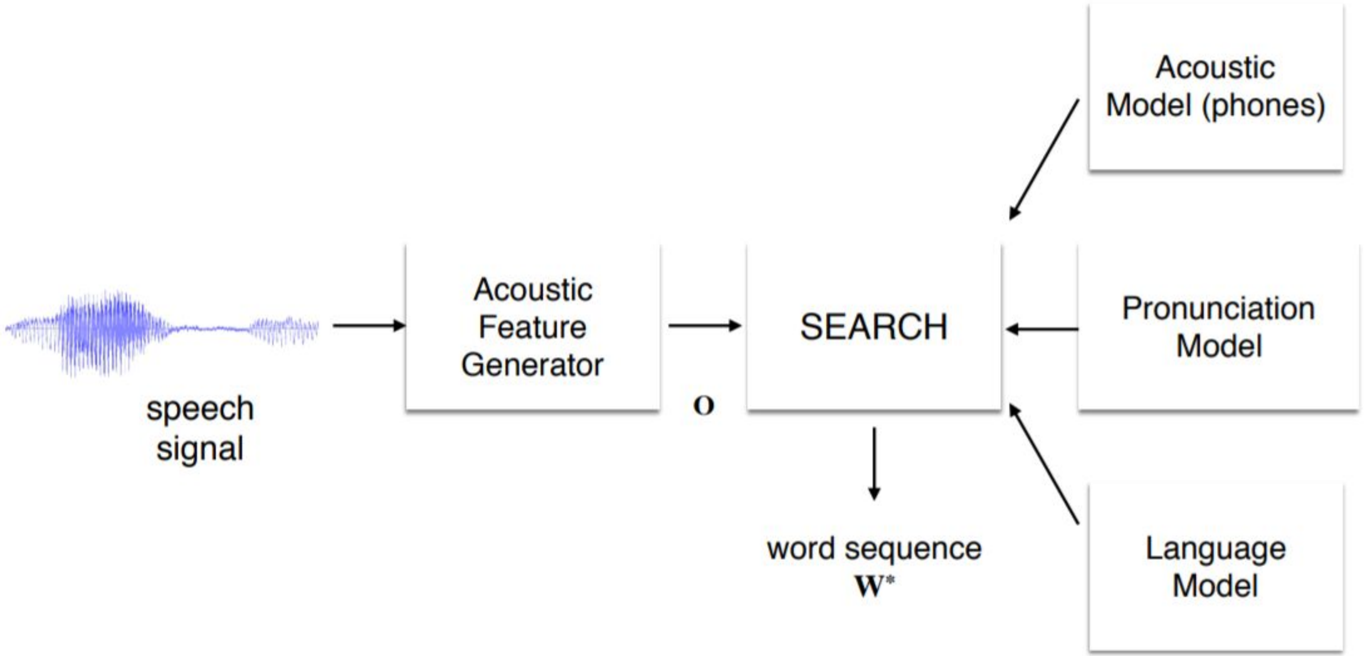


Image source: Lectures on Automatic Speech Recognition by Prof Preethi Jyothi, Course CS753, IIT Bombay

Evaluation Metrics for ASR

- Characteristics of a good ASR Metric:
  - Direct
  - Objective
  - Interpretable
  - Modular

- Word Error Rate: Popular metric; measures the percentage of corrections needed to transform an incorrect word to a correct word.

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C}$$

ASR Error rates across the years

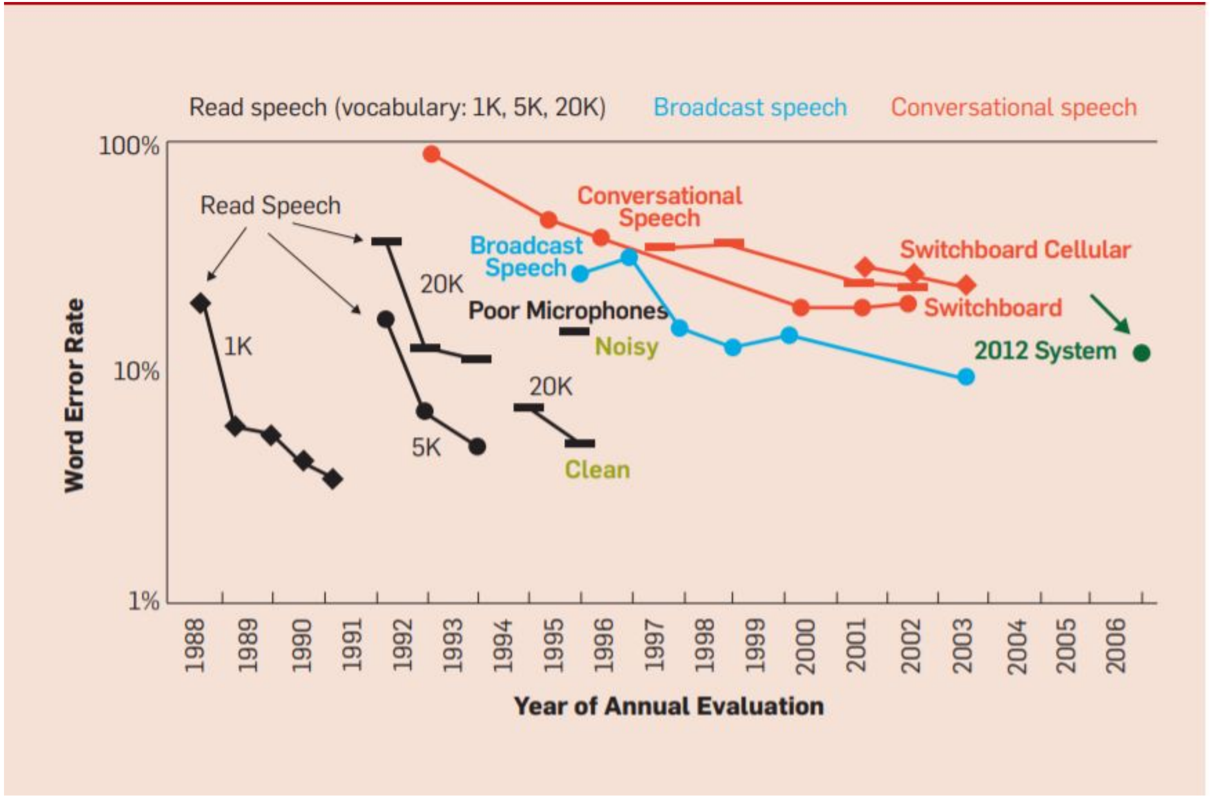


Image Source: A Historical Perspective of Speech Recognition, Communications of the ACM, 2014

Hidden Markov Models

- Each spoken word consists of a sequence of 'l' pronunciation segments called phonemes
- To compute over all possible pronunciations Q of the word, we model the probability as -

$$P(X|W) = \sum_{Q} P(X|Q)P(Q|W)$$

- Each base phoneme q is modelled as a HMM as depicted in the figure below -

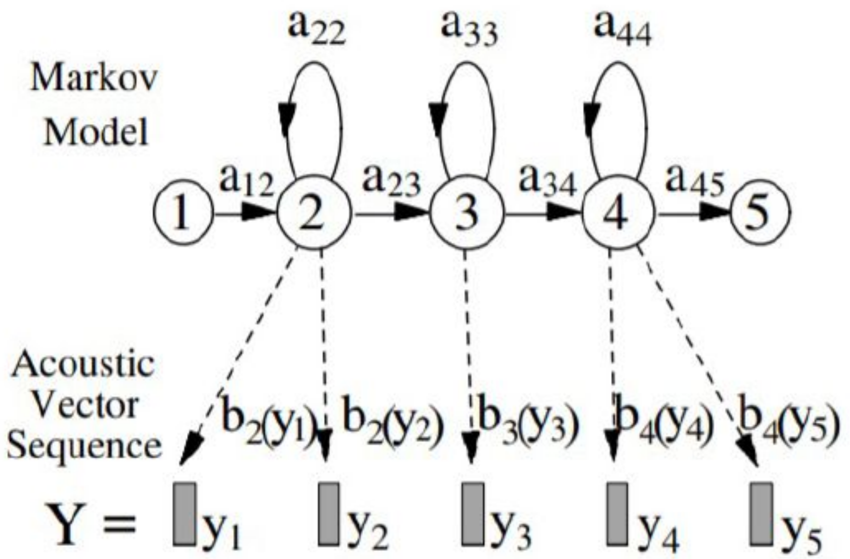


Figure 3.2: Example of a HMM phone model

- For a large vocabulary, including the dependence on all previous words in the sentence becomes computationally difficult task.
- As a result, N-gram models were constructed such that the dependence is limited to the last N words of the sequence -

$$P(W) = \prod_{k=1}^{K} P(w_k | w_1, w_2, \ldots w_{k-N+1}) \qquad P(w_k | w_{k-1}, w_{k-2}) | = \frac{C(w_{k2} w_{k1} w_k)}{C(w_{k2} w_{k1})}$$

Viterbi Algorithm:

- Identifying the sequence of hidden variables given observed sequence
- Calculates the probability of the various possible outputs of the current time step keeping into account the associated probabilities of the previous time step outputs
- At each timestep we have a probability associated with all the possible paths in the decoding.
- The path with the highest probability is assigned as the Viterbi path and is continued forward to the next time step.

A short digress: Viterbi decoding for Part of Speech (POS) tagging

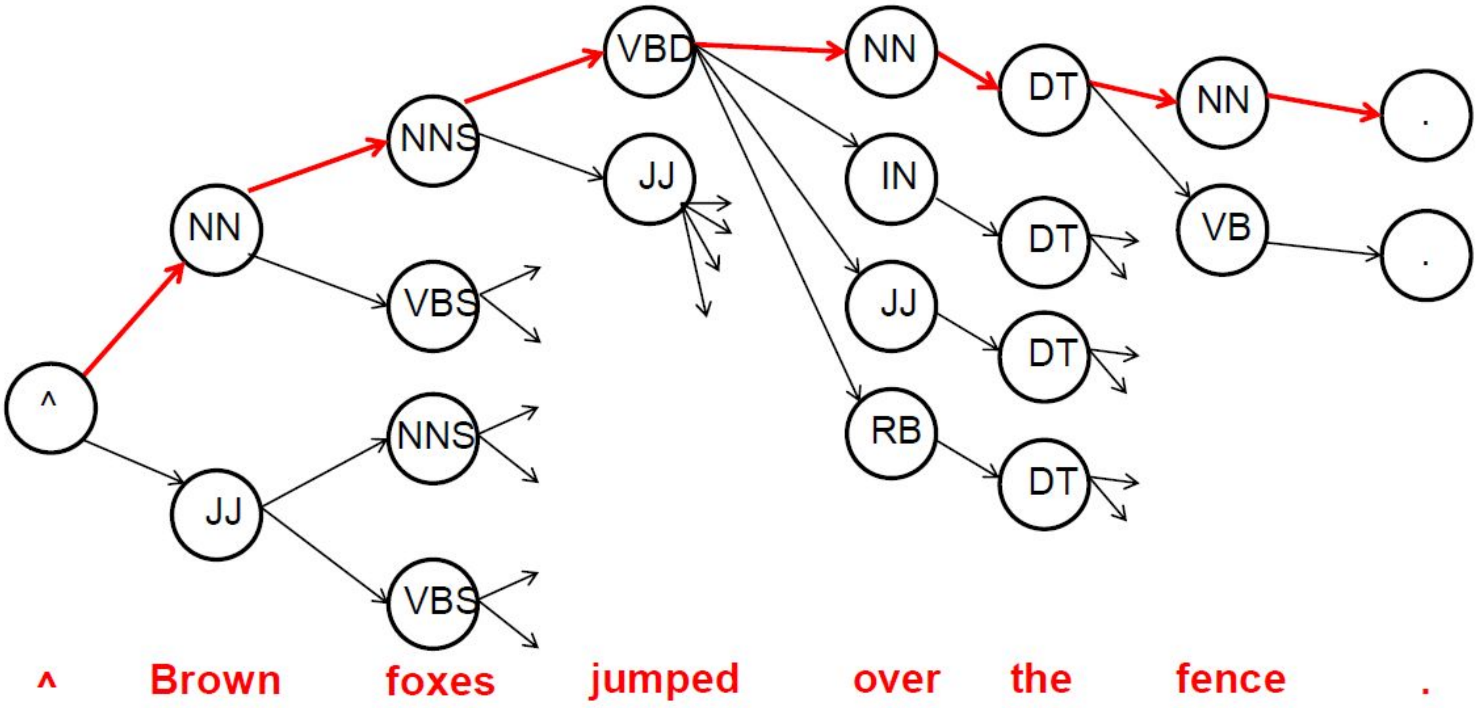


Image source: Lectures on Speech and NLP by Prof Pushpak Bhattacharyya, Course CS626, IIT Bombay

End-to-End Deep Learning Based ASR Systems

Shortcomings of HMM ASR Systems:

- Local optimization of these models does not imply global optimization of the entire pipeline
- Modules are difficult to train and often require a lot of manual and domain specific feature engineering to achieve good results specific to the domain chosen
- Demand a large amount of annotated and aligned speech-text corpus.
- Compounding Errors problem

Demonstrating the power of E2E Models

	dev	test
DNN - HMM	5.0	5.8
E2E (Attention)	14.7	14.7

Results on Librispeech-100 corpus

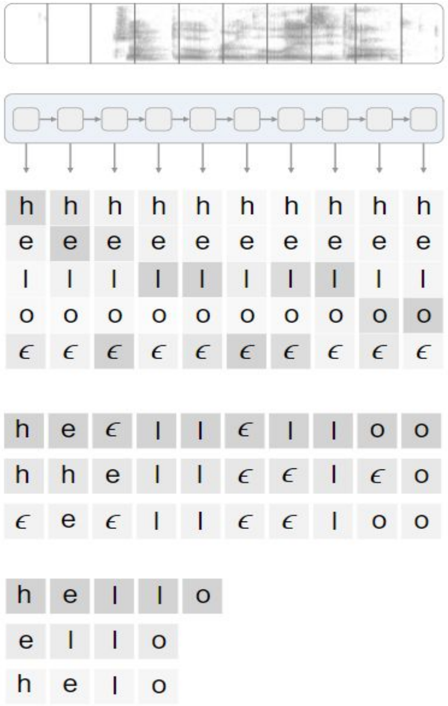
	dev	test
DNN - HMM	4.0	4.4
E2E (Attention)	4.7	4.8

Results on Librispeech-960 corpus

- Three types of E2E systems - 1) CTC-Based Models, 2) RNN Transducer and 3) Attention-Based models

Results taken from “RWTH ASR Systems for LibriSpeech: Hybrid vs Attention”, Christoph et al, 2019

CTC Based ASR Models



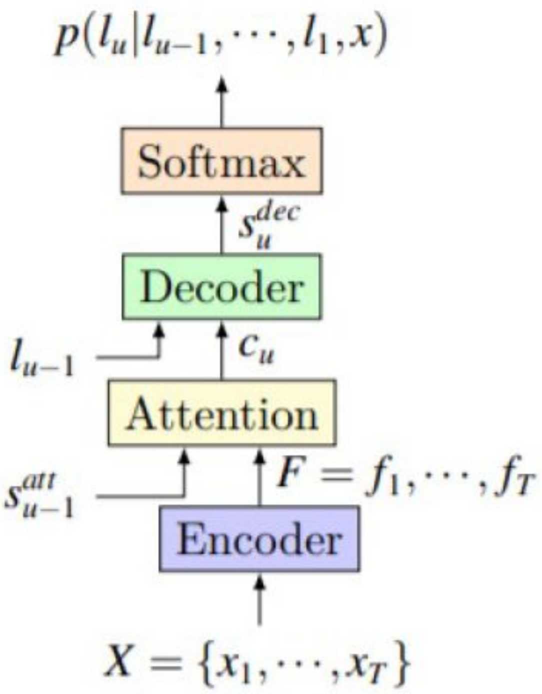
- A loss function but it solves the alignment problem while computing its loss function.
- CTC attempts to solve the data alignment problem as alignment between segments and audio is no longer needed.
- CTC uses the higher dimensional features learnt through DNNs to directly map audio inputs to hypothesis segments.

Image Source: A blog on Sequence Modeling With CTC - https://distill.pub/2017/ctc/

RNN - Transducer Model

- RNN encoder, analogous to an acoustic model, maps input audio features x to hidden representations h^e
- The prediction network takes as input the previous output label prediction w^u and maps it to to a hidden representation h^p
- The joint network takes the encoder representation h^e and the prediction network representation h^p
- Produces joint logits z which are softmax normalized to provide an output distribution over over the output label space and blank symbol

Attention Based ASR Models:



- Encoder model maps the input audio signal to a sequence of vectors instead of one fixed vector
- The decoder then assigns weights to these sequence of vectors while concatenating them to decode the higher dimensional features
- Thus at each time step, the previous as well as the future time step features are taken into account while decoding the particular character and alignment.

Image Source: Deep Context: End-to-End Contextual Speech Recognition, Pundak et al, 2018

Understanding Speech Data

Characteristics of a good ASR dataset:

- Noise Robustness: Presence of sufficiently noisy data for model to learn features with noise
- Diversity: Covering various accents and pronunciations

Bias in ASR:

- ASR unable to recognize and translate the speech impairment and children speech
- Voice assistants perpetuate a racial divide by misrecognising the speech of black speakers more often than of white speakers

Common Pre-processing Steps

Speech:

- Using denoising libraries to reduce background noise by isolating vocals

Text:

- Indian languages Text data must follow ‘utf-8’ encoding
- Removing punctuation marks
- Numbers written in text format

Low Resource ASR

- Low resource languages lack sufficient data to facilitate research and development of intelligent models. Most Indian languages belong to this category
- Recent studies have shown that when DNNs are trained with speech signals from multiple languages, the features learnt are of high quality for downstream tasks.
- These features produce better results for speech recognition compared when the DNN is only trained on the data of a low resource language, which is often extremely less to reach optimization

Citrinet512: Transfer learning in ASR (Part of Nvidia’s NeMo Research) (1/1)

- Citrinet512 is a CTC Based Model
- Contains 1D Convolutional layers with various blocks
- Each block consists of Convolutional layers, Batch Normalization, ReLU and Dropout

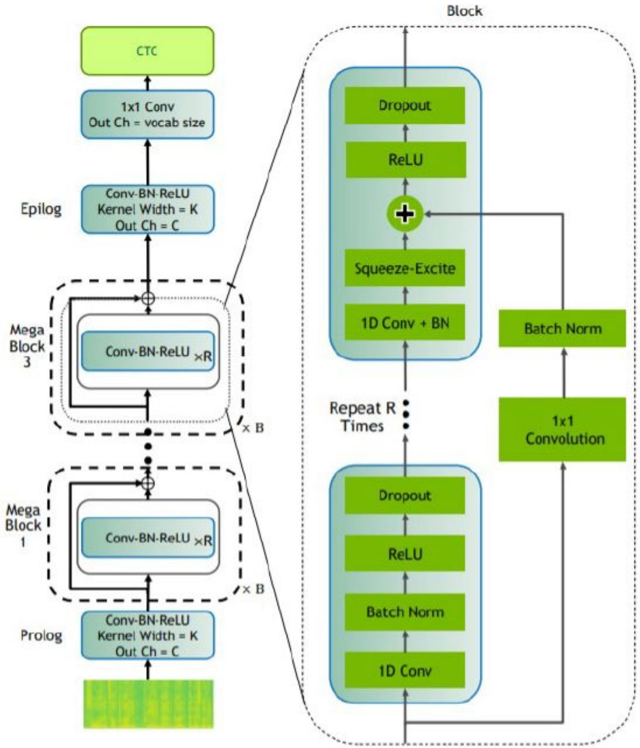


Image Source: Citrinet: Closing the Gap between Non-Autoregressive and Autoregressive End-to-End Models for Automatic Speech Recognition, Majumdar et al, 2021

Citrinet512: Transfer learning in ASR (Part of Nvidia’s NeMo Research) (2/2)

- An additional Squeeze and Excite (SE) layer is also present which improves the representational power of a network by modelling the inter-channel dependencies of convolutional networks.
- Citrinet512 model can be used as a pretrained checkpoint(training on Eng speech) followed by finetuning for Hindi or any other Indian Language
- Finetuning is performed by replacing the vocabulary in the decoder with Devanagiri characters

Wav2vec: Learning Speech representations from Audio

- Uses raw audio to learn speech features
- Masks spans of speech representations
- Training objective is to recover the masked audio through the context

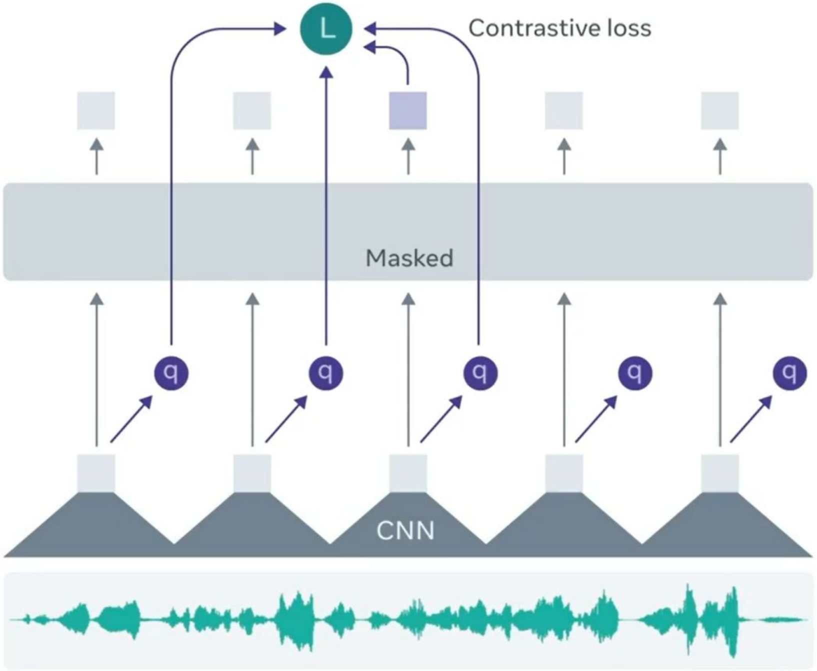


Image Source:
https://ai.facebook.com/blog/wav2vec-20-learning-the-structure-of-speech-from-raw-audio/

Results of wav2vec

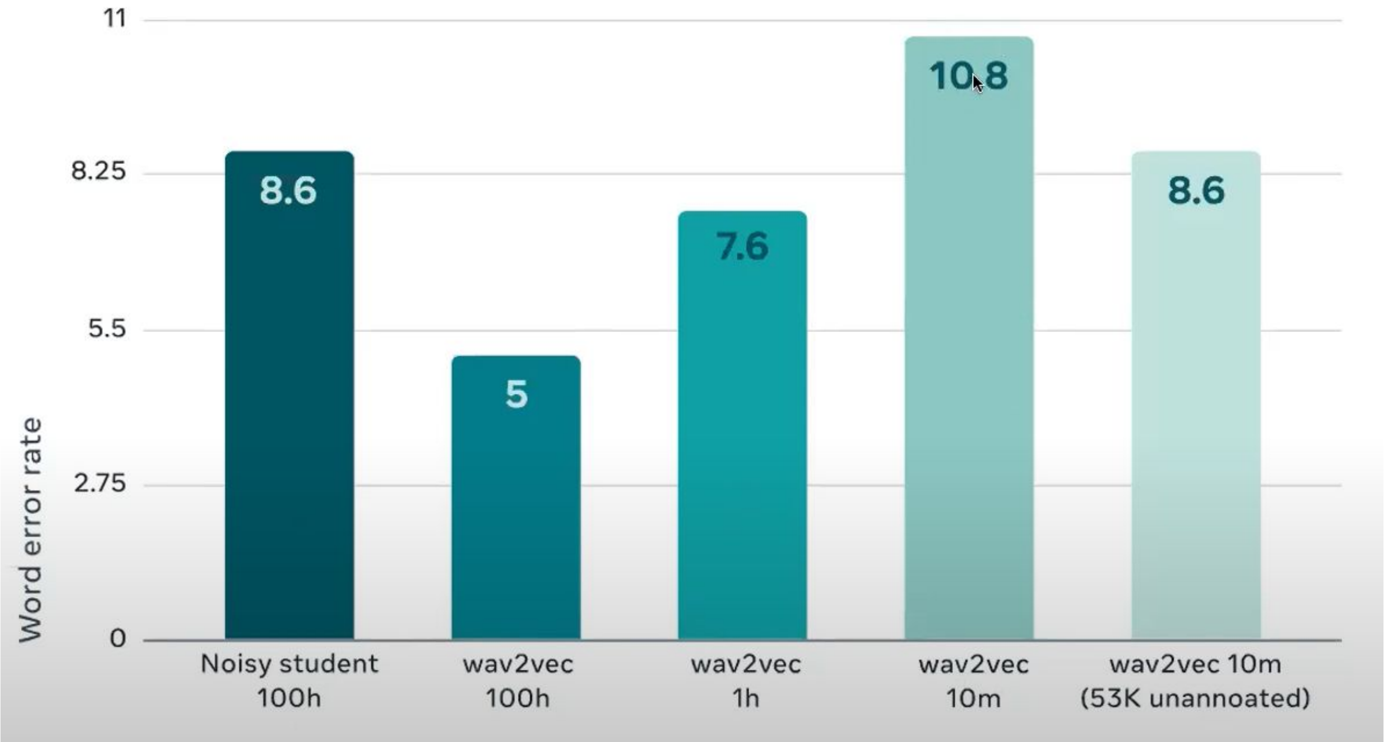


Image Source: https://ai.facebook.com/blog/wav2vec-20-learning-the-structure-of-speech-from-raw-audio/

Finetuning Wav2Vec 2.0 model for multilingual ASR

- The XLSR Wav2Vec2 model is presented by Facebook AI in “Unsupervised Cross Lingual Representation Learning for Speech Recognition“ - Conneau et al, 2020
- XLSR stands for **Cross-Lingual Speech Representations** and refers to XLSR-Wav2Vec2’s ability to learn speech representations that are useful across multiple languages
- XLSR-Wav2Vec2 learns powerful speech representations from hundreds of thousands of hours of speech in more than 50 languages(including Hindi) of unlabeled speech

CLSRIL-23: Pretraining wav2vec 2.0 on Indian Languages ASR Data, July 2021

- Authors use wav2vec 2.0 to pretrain on 9k+ hours of unsupervised speech using 23 Indian Languages followed by finetuning in each of these languages
- Comparison of a monolingual approach and multilingual approach is performed by analysing a model pretrained only on Hindi and a model pretrained on multiple languages

Comparing both the approaches

- Training time for wav2vec 2.0 is much longer than the Citrinet512 model. For wav2vec 2.0, each epoch took about 1.5 hours for training whereas for Citrinet512, it took about 15 min.
  - Possible Reason: wav2vec 2.0 is a transformer based approach whereas Citrinet512 is a convolutional model, performing faster with a GPU
- Ability of the Citrinet512 is able to output punctuation marks much better than wav2vec 2.0

ASR DEMO

Summary of ASR

- Historical view of the ASR problem statement and the progress made in the last 70 years
- Understanding the traditional HMM models, its strengths and its weaknesses and how it paved the way for E2E systems
- Appreciating the state-of-the-art E2E system for ASR: wav2vec and multilingual training for low resource ASR
- Demo for English and Hindi ASR using this SOTA

Disfluency Correction

Disfluency

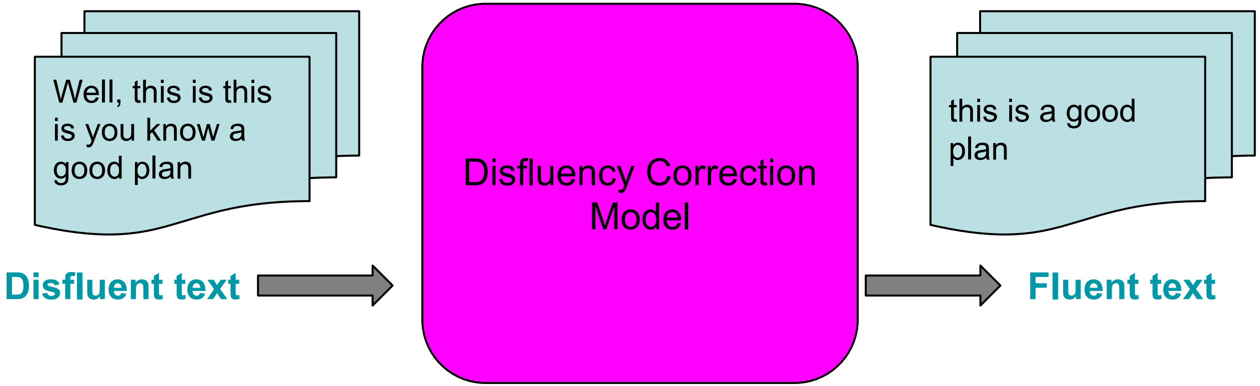
Speakers often **use filler words**, **repeat fluent phrases**, suddenly **change the content of speech**, and **make corrections** in their statement. These are some common disfluencies. Also, speakers use words like ”yeah”, ”well”, ”you know”, ”alright” etc. which do not contribute to the semantic content of the text but are only used to fill pauses or to start a turn, are considered to be disfluent.

Example: Well, this is this is you know a good plan.

Motivation

- Disfluency is a characteristic of spontaneous speech which is not present in written texts
- It reduces the readability of speech transcripts
- It also poses a major challenge to downstream tasks e.g. machine translation
- Since MT models are usually trained on fluent clean corpora, the mismatch between the training data and the actual use case decreases their performance.
- To tackle this challenge, specialized disfluency correction models are developed and applied as a post-processing step to remove disfluencies from the output of speech recognition systems

Disfluency Correction



Types of Disfluencies (1/2)

Type	Description	Example
Filled Pause	Non lexicalized sounds with no semantic content.	but **uh** we have to go through the same thing.
Interjection	A restricted group of non lexicalized sounds indicating affirmation or negation.	**uh-huh,** I can understand the issue.
Discourse Marker	Words that are related to the structure of the discourse in so far that they help beginning or keeping a turn or serve as acknowledgment. They do not contribute to the semantic content of the discourse.	**Well,** this is a good plan.

Types of Disfluencies (2/2)

Type	Description	Example
Repetition or Correction	Exact repetition or correction of words previously uttered.	**If** if I **can't** don't know the answer myself, I will find it.
False Start	An utterance is aborted and restarted with a new idea or train of thought.	**We'll never find a day** what about next month?
Edit	Phrases of words which occur after that part of a disfluency which is repeated or corrected afterwards or even abandoned completely, to indicate that the previously uttered words are not intended.	We need two tickets, **I'm sorry,** three tickets for the flight to Boston.

Disfluency: Surface Structure (1/2)

- Reparandum
  - contains those words, which are originally not intended to be in the utterance
- Interruption point
  - marks the end of the reparandum
- Interregnum
  - consists of an editing term or a filler
- Repair
  - words from the reparandum are finally corrected or repeated here or a complete new sentence is started

Disfluency: Surface Structure (2/2)

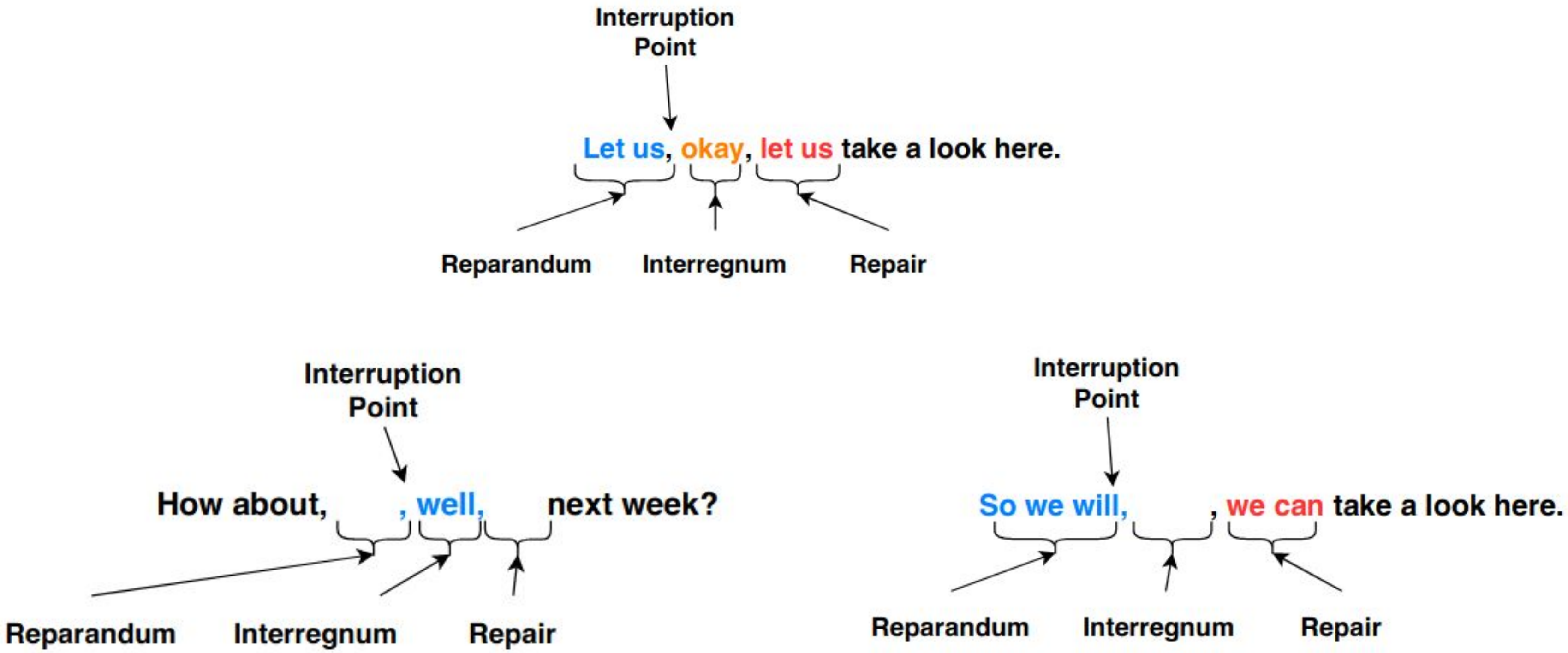


Image source: Saini et al. [4]

Dataset (1/2)

SwitchBoard English Disfluency Correction corpus.

Split	No of Sentence
Train	55482
Validation	11889
Test	11889

Dataset (2/2)

SwitchBoard English Disfluency Correction corpus.

Disfluent Sentence	Fluent Sentence
well i i just live in a i live in an apartment now	i live in an apartment now
it's a it's a fairly large community	it's a fairly large community
and so uh you know i'm kind of spoiled	i'm kind of spoiled

Data Preprocessing

1. Lower-case
2. Normalization
3. Punctuation removal
4. Apply Byte-Pair Encoding (BPE)

Evaluation

- BLEU score
- Precision, Recall, F1 score (for disfluency detection)
- Human evaluation
  - Disfluent words are dropped
  - Fluent words are retained

$$precision = \frac{true\ positive}{true\ positive + false\ positive} \quad recall = \frac{true\ positive}{true\ positive + false\ negative}$$

$$F1\ score = \frac{2 * precision * recall}{precision + recall}$$

Sequence to Sequence

Model the problem as a Translation task as if

Disfluent Sentence: Sentence in language 1

Fluent Sentence : Sentence in language 2

Performance on SWBD Testset: 95.01 bleu score

Usage of Joint BPE

- Using different BPE vocabulary at source and target sides, model achieved 94.70 bleu score on test set
- Use of joint BPE vocabulary (from the concatenated corpus of disfluent sentences and fluent sentences) improved the performance to 95.01

Style Transfer (1/3)

- Unsupervised: works without any parallel data (fluent, disfluent pairs)
- For every mini-batch of training, soft translations for a domain are first generated
- Subsequently they are translated back into their original domains to reconstruct the mini-batch of input sentences.
- The sum of token-level cross-entropy losses between the input and the reconstructed output serves as the reconstruction loss.

Style Transfer (2/3)

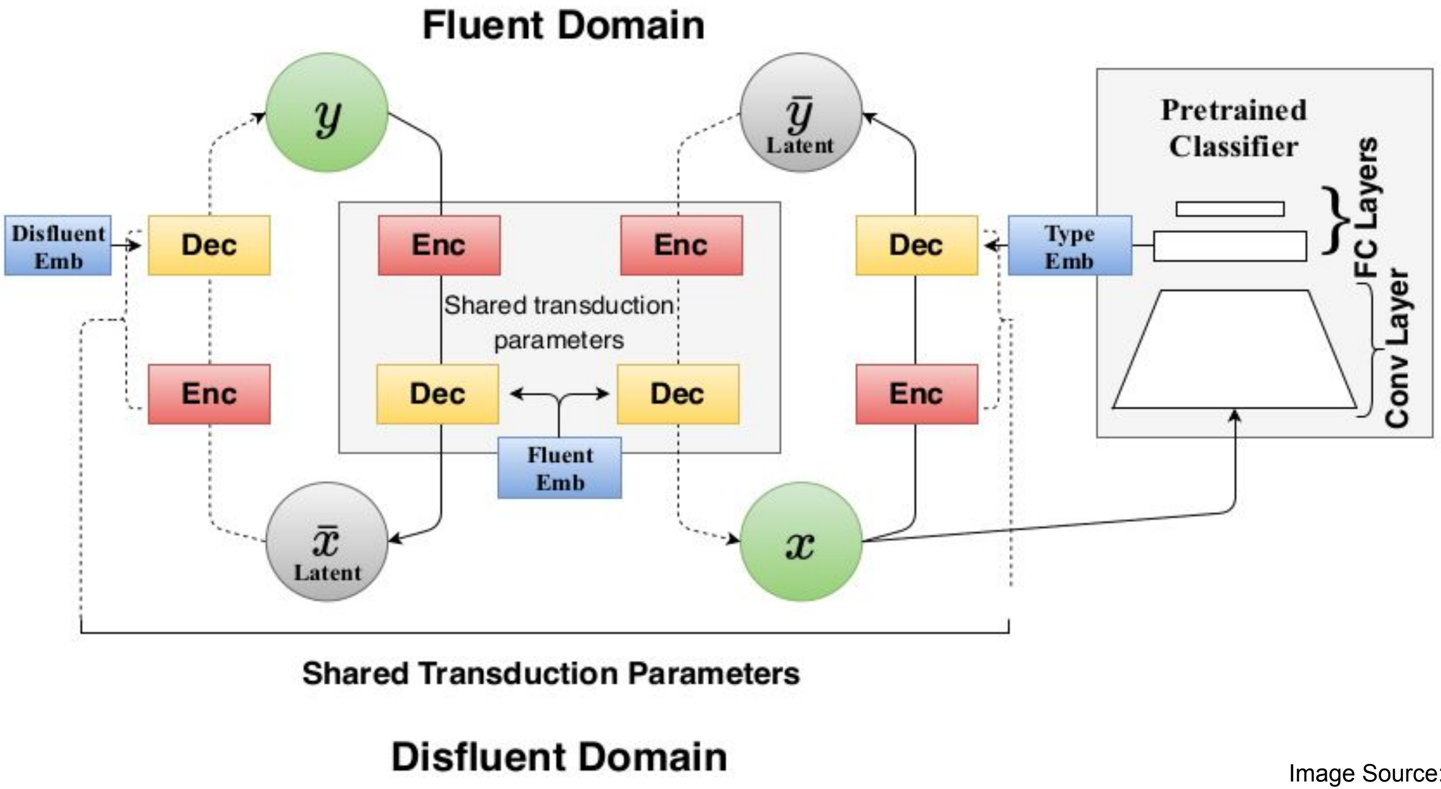
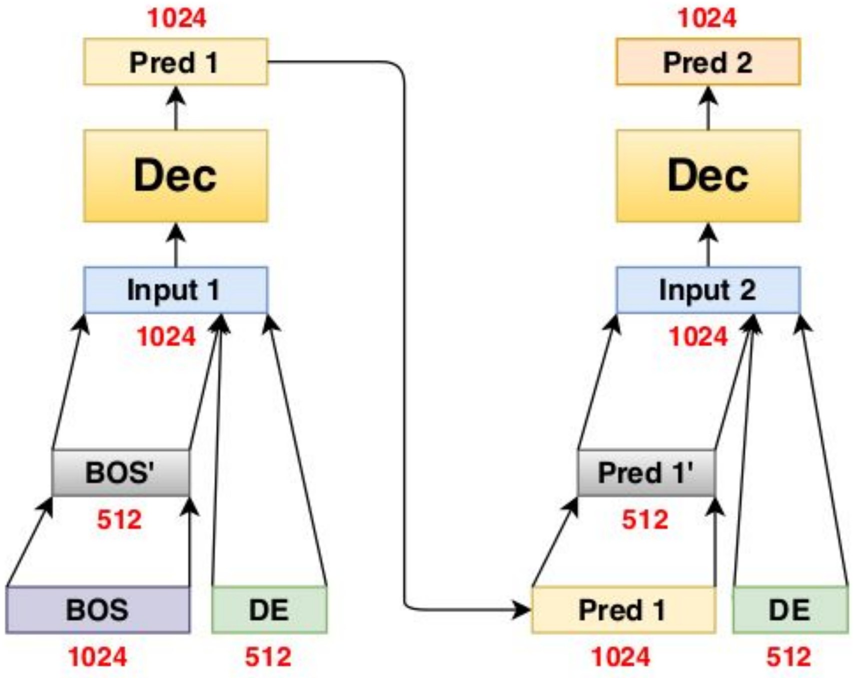


Image Source: Saini et al. [3]

Style Transfer (3/3)

- It consists of a single encoder and a single decoder
- Since we are only operating on the English language in the source (disfluent) and target (fluent), it is important to utilize the benefit of parameter sharing.
- Single encoder and single decoder are used to translate in both directions, i.e., from disfluent to fluent text and vice-versa.
- The decoder is additionally conditioned using a domain embedding to convey the direction of translation, signifying whether the input to the encoder is a fluent or disfluent sentence.

Domain Embedding



Dimensionality reduced word embedding is concatenated with the domain embedding DE at every time-step(t) to form the input for the decoder.

Image Source: Saini et al. [3]

Disfluency Correction using Unsupervised and Semi-supervised Learning: Results

#Sentences	Percentage(%)	Dev	Test
0	(Unsupervised) 0	78.72	79.39
554	1	83.85	85.28
2774	5	84.67	86.03
5548	10	84.98	86.12
13870	25	85.88	87.04
27741	50	86.10	87.90
55482	100	87.16	88.22

Image Source: Saini et al. [3]

Disfluency Detection

- Detect the disfluent words in the sentences
- Remove those disfluent words from the disfluent sentence
- We would get the fluent sentence

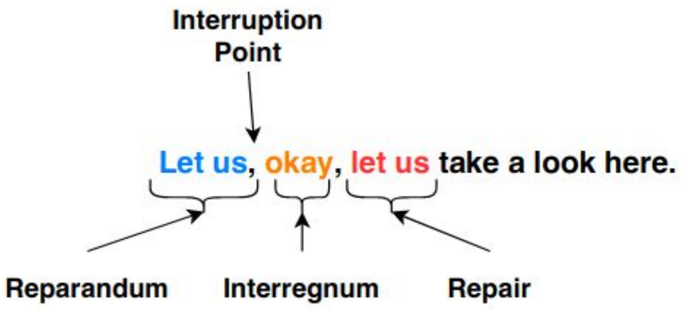


Image Source: Saini et al. [4]

Disfluency Removal

- Detect the disfluent words in the sentences
- Remove those disfluent words from the disfluent sentence
- We would get the fluent sentence

Disfluent: this is this is you know a big problem
Fluent: this is a big problem

Sequence Tagging

- We tag each of the words as disfluent or fluent.
- 0 corresponds to fluent and 1 corresponds to disfluent.
- Train model on the sequence tagging task

Disfluent: this is this is a big problem

Fluent: this is a big problem

Tags: 1 1 0 0 0 0 0

Synthetic Data: Motivation

- Disfluency phenomenon is clearly visible in the Indian languages.
- But due to the unavailability of disfluency correction dataset, it is not possible to train models.
- This motivates the work of disfluency correction without any real parallel data.

Rule-based Disfluency Generation

Pronoun phrase repetition

- Original fluent sentence: i was saying that we should go for a movie
- Disfluent sentence: *i was* i was saying that we should go for a movie

Insertion of filler words

- Original sentence: the new year is looking grim
- Disfluent sentence: *ah ah* the new year is looking grim

Example Source: https://aclanthology.org/2020.iwslt-1.22.pdf

Few Rules for Bengali (1/6)

Randomly insert frequent filler words

- Bn: বাপু-র নেতৃত্বে **মানে** পরিচালিত ঐতিহাসিক জন-আন্দোলন 'চম্পারনসত্যাগ্রহ'-এর প্রভাব ছিল সুদূর প্রসারিত।
- Transliteration: vApu-ra netRRitve mAne parichAlita aitihAsika jana-Andolana 'champAranasatyAgraha'-era prabhAva Chila sudUra prasArita|
- En: The historical mass movement ’Champaran Satyagraha’, I mean conducted under the leadership of Bapu, had a far-reaching effect.

Few Rules for Bengali (2/6)

Word Repetition

- Bn: সেজন্যই আমরা আপনাদের সহযোগিতায় দেশের সকল জমির জন্য 'মৃত্তিকা স্বাস্থ্য কার্ড' **চালু** চালু করার অভিযান শুরু করেছি।
- Transliteration: sejanyai AmarA ApanAdera sahayogitAya deshera sakala jamira janya 'mRRittikA svAsthya kAr.Da' chAlu chAlu karAra abhiyAna shuru kareChi|
- En: That is why we have started the campaign to introduce introduce 'Soil Health Card' for all the lands of the country with your cooperation.

Few Rules for Bengali (3/6)

Synonym correction

- Bn: আজ আমরা **খবরের কাগজের** সংবাদপত্রের কথায়- সশস্ত্র বাহিনীকে তাদের ইচ্ছামতো কাজের পূর্ণ স্বাধীনতা দিয়েছি।
- Transliteration: Aja AmarA khavarera kAgajera saMvAdapatrera kathAya়- sashastra vAhinIke tAdera ichChAmato kAjera pUrNa svAdhInatA diya়eChi|
- En: Today, in the words of the news paper newspaper, we have given the armed forces full freedom to do whatever they want.

Few Rules for Bengali (4/6)

Missing syllables

- Bn: ওড়িশার সার্বিক উন্নয়নে কেন্দ্রীয় সরকারের **প্রতিশ্রুতিদ্ধতার** প্রতিশ্রুতিবদ্ধতার কথা পুনর্ব্যক্ত করেন প্রধানমন্ত্রী।
- Transliteration: o.Da◌়ishAra sArvika unnaya◌়ne kendrIya◌় sarakArera pratishruti~~va~~ddhatAra pratishrutivaddhatAra kathA punarvyakta karena pradhAnamantrI|
- En: The Prime Minister reiterated the central government's commitment to the overall development of Orissa.

Few Rules for Bengali (5/6)

Pronoun Correction

- Bn: সন্ত্রাসবাদের হুমকি যেভাবে দিন দিন বেড়ে চলেছে, **তাকে না** তার মোকাবিলায় সম্ভাব্য পদক্ষেপ গ্রহণের বিষয়গুলি সম্পর্কেও আলোচনা করেন দুই প্রধানমন্ত্রী।
- Transliteration: santrAsavAdera humaki yebhAve dina dina ve.Da়e chaleChe, tAke nA tAra mokAvilAya় sambhAvya padakShepa grahaNera viShaya়guli samparkeo AlochanA karena dui pradhAnamantrI|
- En: The two Prime Ministers also discussed the possible steps to be taken to counter the growing threat of terrorism.

Few Rules for Bengali (6/6)

Use part of word before the actual word

- Bn: কিন্তু **প্রোমো** প্রোমোটারচক্রের ফাঁদে পড়ে ঠকে যান।
- Transliteration: kintu promo promoTArachakrera phA.Nde pa.Da়e Thake yAna|
- En: But fall into the trap of the promo promoter cycle.

Summary

- Disfluency correction is a crucial step in SSMT pipeline
- It removes irregularities from speech transcriptions and make that ready for Machine Translation
- We are able to build DC systems with parallel data or monolingual data (in both domains) along with little amount parallel data
- Model trained on artificially generated data has limitations
- It would be interesting to take help from other languge's disfluency correction data

Disfluency Correction Demo

https://www.cfilt.iitb.ac.in/speech2text/

References

[1] Matthias Honal and Tanja Schultz. 2003. Correction of Disfluencies in Spontaneous Speech using a Noisy-Channel Approach. In Interspeech.

[2] John Godfrey and Edward Holliman. 1993. Switchboard-1 release 2 LDC97S62. Published by: Linguistic Data Consortium, Philadelphia, USA.

[3] Nikhil Saini, Drumil Trivedi, Shreya Khare, Tejas Dhamecha, Preethi Jyothi, Samarth Bharadwaj, and Pushpak Bhattacharyya. 2021. "Disfluency Correction using Unsupervised and Semi-supervised Learning." In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pp. 3421-3427.

[4] Nikhil Saini, Preethi Jyothi and Pushpak Bhattacharyya, Survey: Exploring Disfluencies for Speech To Text Machine Translation.

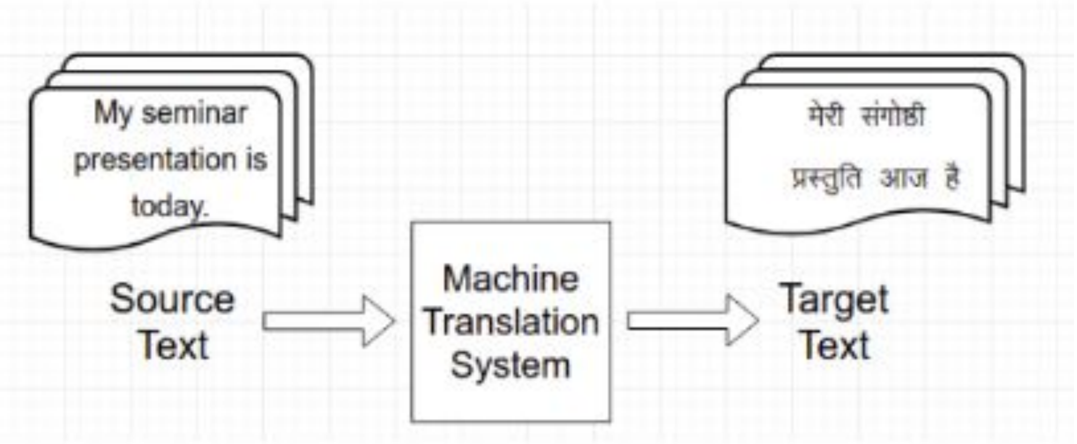
[5] Vicky Zayats, Mari Ostendorf, and Hannaneh Hajishirzi. 2016. Disfluency detection using a bidirectional LSTM. CoRR, abs/1604.03209

Machine Translation

Content

What is Machine Translation?

- Automatic conversion of text from one language to another
  - Preserve the meaning
  - Fluent output text



History of MT

- 1954: First public demo of MT by IBM
  - Georgetown IBM experiment
- 1956: First MT conference
- 1972: Logos MT system
  - Translating military manuals into Vietnamese
  - Rule based approach
- 1993: Statistical MT
  - IBM models
- 2013: Neural Machine Translation

Why MT is hard?

Why MT is hard?

Language Divergence

Language divergence

- Languages express meaning in divergent ways
- **Syntactic divergence**
  - Arises because of the difference in structure
- **Lexical semantic divergence**
  - Arises because of semantic properties of languages

Different kinds of syntactic divergence

- Constituent order divergence (Word order)

English: He is waiting for him.
Hindi: वह उसके लिए इंतजार कर रहा है।

Subject	He	वह
Verb	waiting	इंतजार कर रहा है
Object	him	उसके

- Adjunction divergence

English: Delhi, the capital of India, has many historical buildings.
Hindi: भारत की राजधानी दिल्ली में बहुत सी एतिहासिक इमारतें हैं

- Null subject divergence

English: I am going.
Hindi: जा रहा हूँ।।

Different kinds of lexical semantic divergence

- Conflational divergence

English: He stabbed him.
Hindi: उसने उसे छुरे से मारा

- Categorial divergence (Lexical category change)

English: They are competing.
Hindi: वे प्रतिस्पर्धा कर रहे हैं

- Head-swapping divergence (Promotion or demotion of logical modifier)

English: The play is on.
Hindi: खेल चल रहा है

The Vauquois Triangle

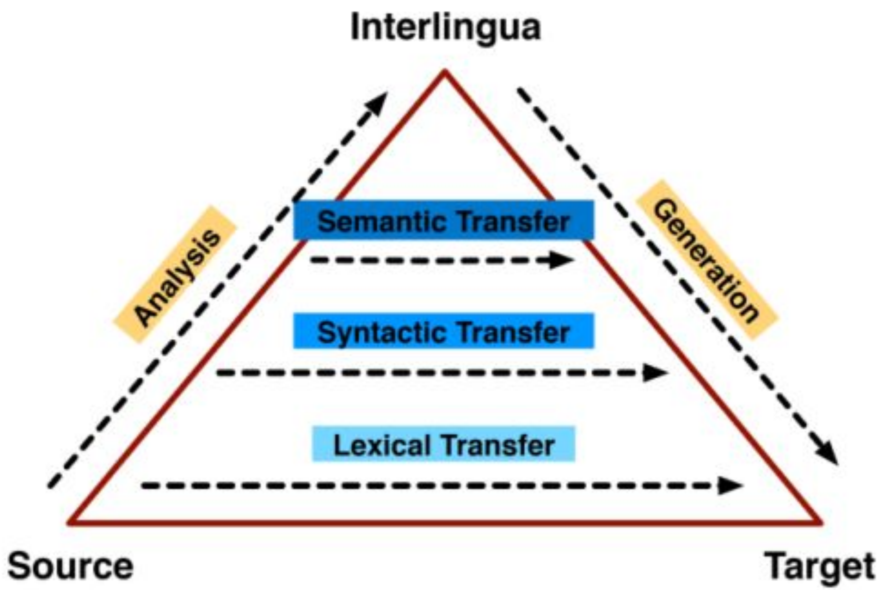


Image source: http://www.cs.umd.edu/class/fall2017/cmsc723/slides/slides15.pdf

MT Paradigms

Different paradigms of Machine Translation

- Rule based Machine Translation
- Statistical Machine Translation
- Example based Machine Translation
- Neural Machine Translation

Rule based Machine Translation

- Linguists create rules
- Three types
  - Direct
    - Map input to output with basic rules
  - Transfer based
    - Direct + Morphological and Syntactic analysis
    - The level of transfer is dependent on the language pairs
  - Interlingua based
    - Use an abstract meaning
    - Interlingua: Represent meaning of text unambiguously
    - It works at the highest level of transfer
- Performance of system highly dependent on experts who are creating rules

Statistical Machine Translation

- Learning from parallel corpora
- Three important things
  - Word translation
  - Word alignment
  - Word fertility management
- Problem to solve for SMT

$$\hat{e} = \arg\max_{e} \left(P(e|f)\right) = \arg\max_{e} \left(P(e).P(f|e)\right)$$

e is target language sentence, *f* is source language sentence, *P(e)* is language model in target language and *P(f|e)* is translation model.

Example based Machine Translation

- Focus is on: Analogy
- Based on textual similarity
- Process
  - Analysis
    - Phrasal fragments of the input sentence
  - Transfer
    - Finding the aligned phrases from the database of examples
  - Generation
    - Recombination (Stitch together the aligned phrases)

Example based Machine Translation: Example

- He buys a book on Machine Translation.
- Phrasal fragments: He buys, a book, on, Machine Translation
- Aligned phrases: Identifies the aligned phrases from the database

He buys: वह खरीदता है

a book: एक पुस्तक

on: पर

machine translation: मशीन अनुवाद

- Recombination: Recombine those phrases to construct a sentence (Adjusting morphology, reordering)

वह मशीन अनुवाद पर एक पुस्तक खरीदता है।

Phrase based Statistical Machine Translation

- Why?
  - Translation of phrases is more intuitive
- Process involved
  - Two-way alignment
    - Using SMT (eg. IBM model 1)
  - Symmetrization
  - Expansion of aligned words to phrases (Phrase table construction)

Phrase based SMT: English to Hindi alignment

	वह	आज	शाम	को	केक	बनाने	की	योजना	बना	रहा	है
He	✓										
is											✓
planning								✓			
to											
make						✓					
a											
cake											
in											
the											
evening			✓								

Phrase based SMT: Hindi to English alignment

	He	is	planning	to	make	a	cake	in	the	evening
वह	✓									
आज										
शाम										✓
को										
केक							✓			
बनाने					✓					
की										
योजना			✓							
बना										
रहा										
है		✓								

Phrase based SMT: Phrase generation

	वह	आज	शाम	को	केक	बनाने	की	योजना	बना	रहा	है
He	✓										
is											✓
planning								✓			
to											
make						✓					
a											
cake					✓						
in											
the											
evening			✓								

- Principle of coverage: Every word must be in a phrase
- Principle of non-vacuousness: No empty phrases
- Principle of consistency:The aligned phrases must be consistent in the sense all words of phrase in source languages

MT Evaluation

- Manual evaluation
- Quality of sentence depends on two factors
  - Adequacy
    - How faithful the meaning of a sentence is transferred
  - Fluency
    - Acceptability of the native speaker

More fluent: मुझे भूख लग रही है।

Less fluent: मैं भूखा महसूस कर रहा हूँ।

- Automatic evaluation measures
  - Word/phrase matching based
  - Edit distance based
  - Ranking based

BLEU score

- Bilingual Evaluation Understudy
- Word/Phrase matching based

$$BLEU = BP.exp(\sum_{n=1}^{N}(w_n.log(p_n))$$

- BP is brevity penalty, to penalize based on the length of the generated sentence.

$$BP = \begin{cases} 1 & c > r \\ e^{(1-r/c)} & c <= r \end{cases}$$

c = the length of the candidate translation, r = the effective reference corpus length, p_n is modified n-gram precision, w_n is weight(uniform in BLEU)

BLEU score: Example

English: He is a painter.

Hindi (Candidate): वह एक चित्रकार चित्रकार चित्रकार है।

Hindi (Reference): वह एक चित्रकार है।

- Example:
  - 1-gram precision is 1.
  - Modified 1-gram precision is 4/6.
- The ratio of the number of phrases of length n present in candidate translation that are also present in reference translation and total number of phrases of length n in candidate translation.
- In modified n-gram precision maximum count from reference translation

Neural Machine Translation

Why Neural Machine Translation?

- In RBMT, EBMT, and SMT, there is no notion of similarity or relationship between symbolic representation of individual words.
- Ability to translate *I go to school* does not make these models capable of translating *I went to college*.
- However, Neural Network techniques work with distributed representations.
- NMT evaluates a single formula that explains all rules of the translation task. (Generalisation)

Paradigms of Machine Translation

Human effort ←

Data hungry →

RBMT → EBMT → SMT → NMT

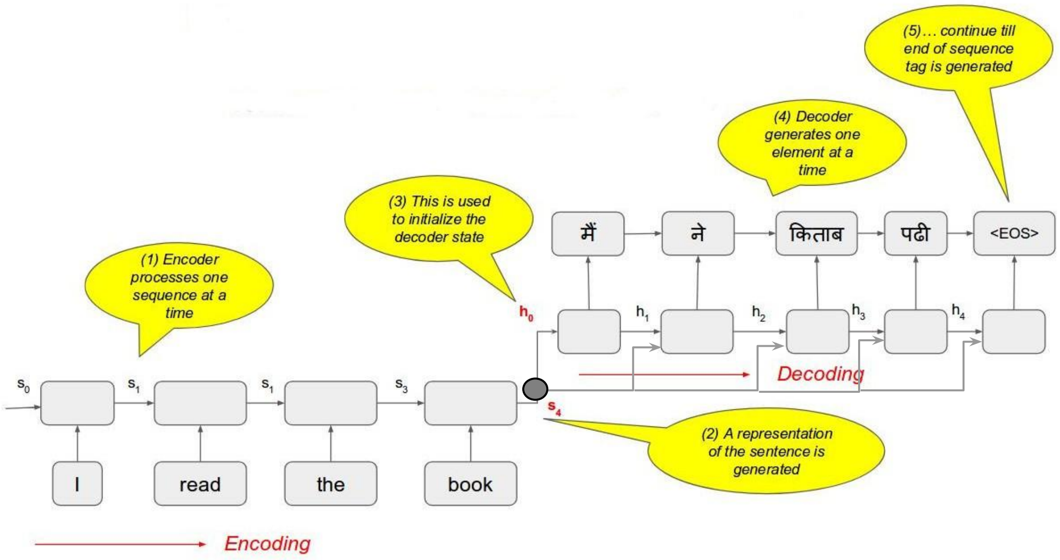
(Rule based machine translation) (Example based machine translation) (Statistical machine translation) (Neural machine translation)

Rules handmade by human expert → Rules by learning from data

What is NMT?

- The task of MT is a sequence-to-sequence problem.
- It uses an encoder-decoder NN architecture with attention mechanism.
- NMT requires large parallel corpus.
- Here, we will discuss RNN-based and Transformer-based encoder-decoder architectures.

Simple RNN-based Encoder-Decoder [9] architecture overview



s_4 is the summary vector

Image source - [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Summary vector representation

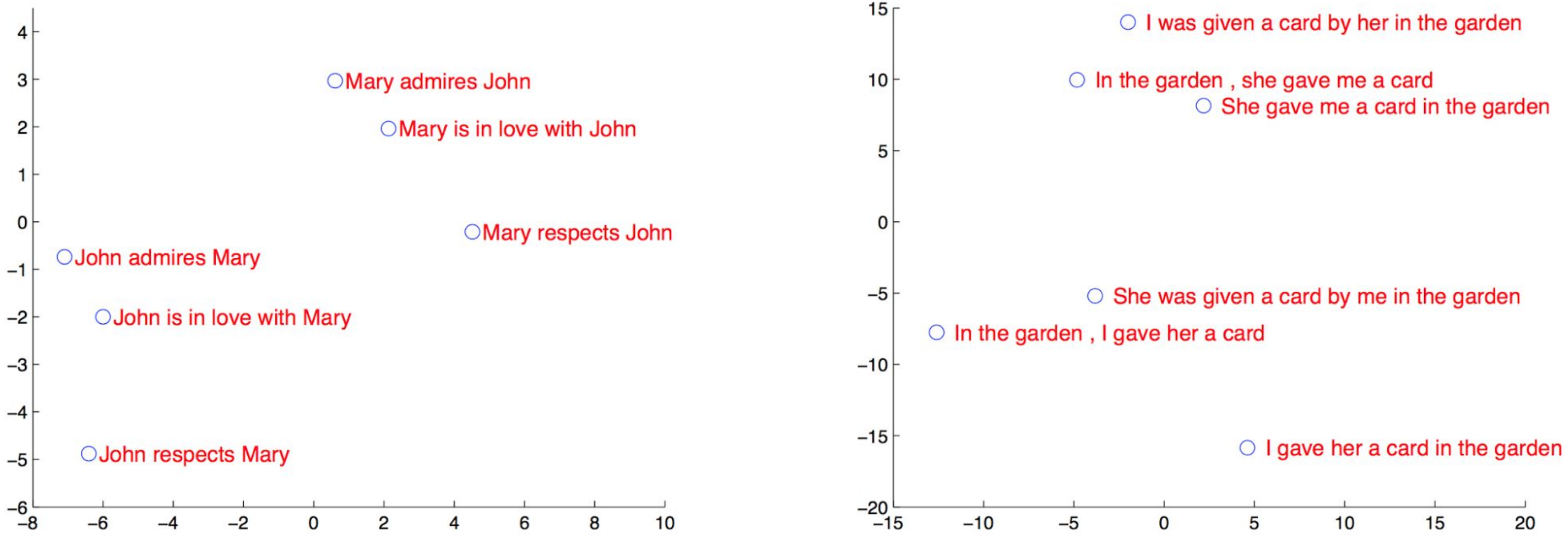


Image source- [9]

Problems with simple Encode-Decode paradigm (1/2)

What happens in enc-dec architecture-

1. Encoding transforms the entire sentence into a single vector.
2. Decoding process uses this sentence representation for predicting the output.

Problems-

- Quality of prediction depends upon the quality of sentence embeddings.
- After few time-step, summary vector may lose information of initial words of input sentence.

Problems with simple Encode-Decode paradigm (2/2)

Possible solutions-

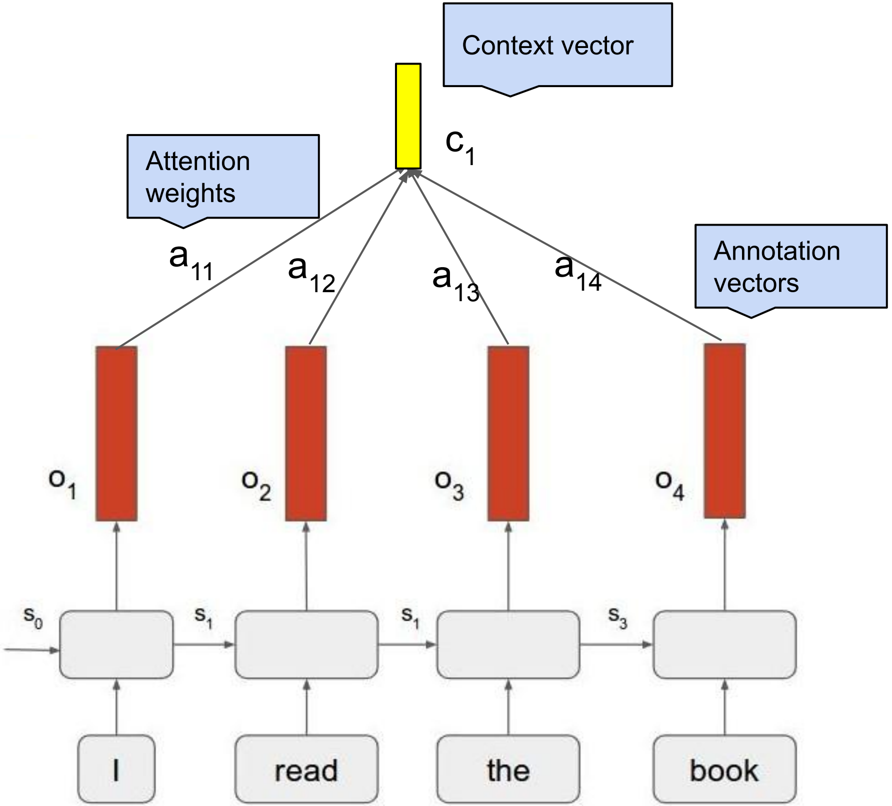
- For prediction at each time step, present the representation of the relevant part of the source sentence only.

the girl goes to school

लड़की स्कूल जाती है

  - Attention-based encoder-decoder

Annotation vectors and context vectors



Attention weights are calculated from alignment scores which are output of another feed-forward NN which is trained jointly.

Attention-based Encoder-Decoder [10] architecture

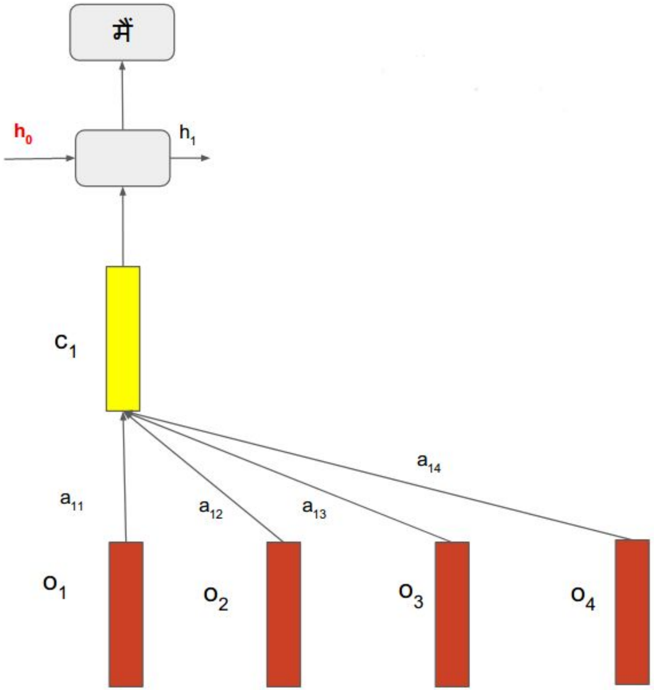


Image source- [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Attention-based Encoder-Decoder [10] architecture

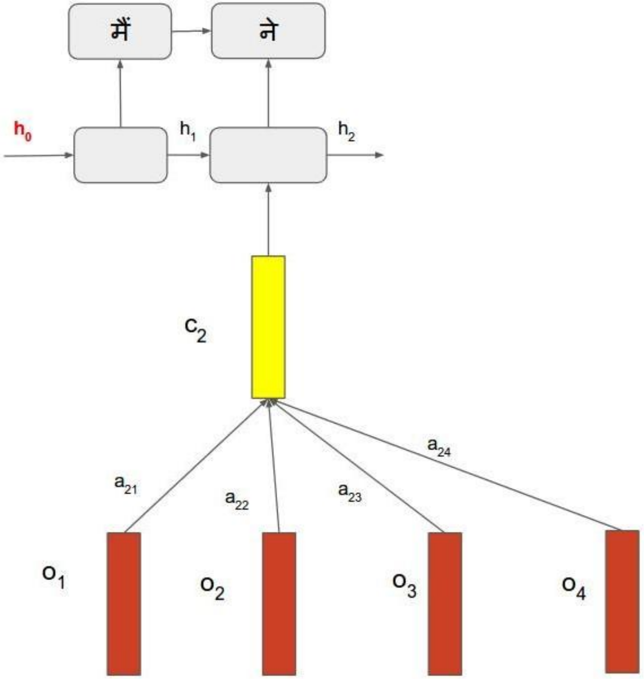


Image source- [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Attention-based Encoder-Decoder [10] architecture

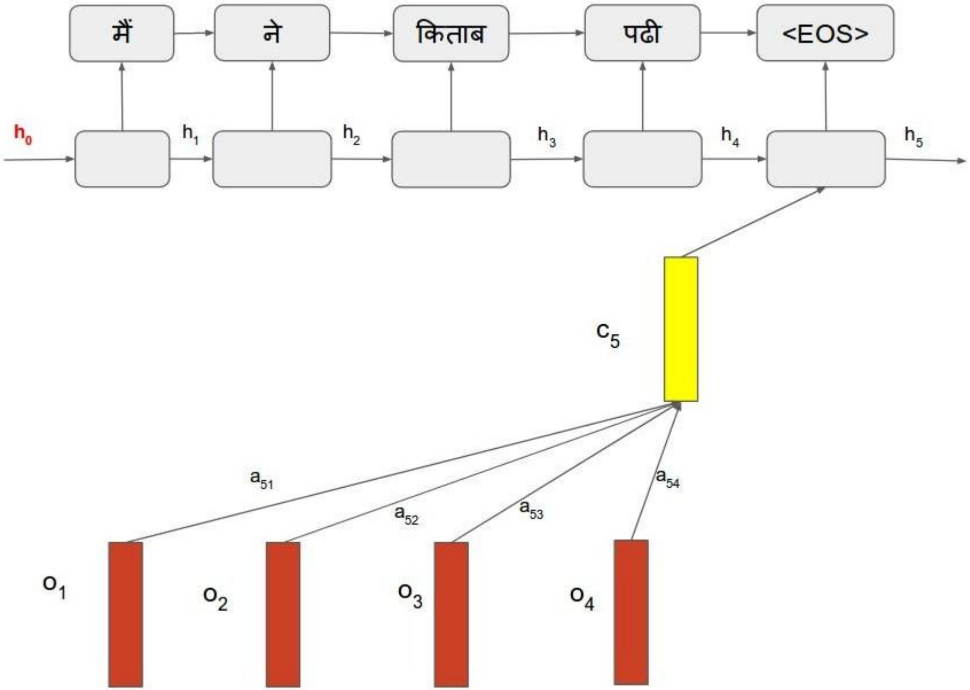


Image source- [http://www.iitp.ac.in/~shad.pcs15/data/nmt-rudra.pdf]

Transformer [11]

- Motivations to choose Transformer over RNN-
  - Faster
  - More efficient.
- Architecture-
  - This is an encoder-decoder architecture with Transformers instead of RNNs.

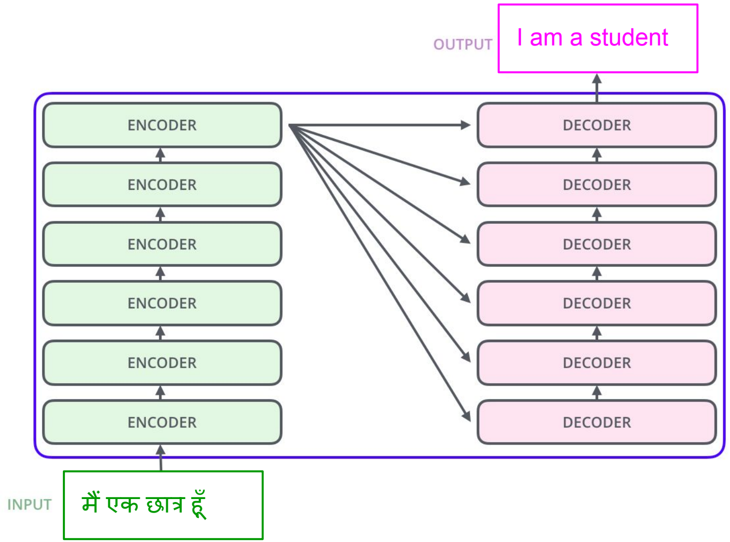


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Embedding

- Embedding-
  - Input of the encoder = sum(word_embedding, positional encoding)
  - To set a constant and small vector_size of positional encoding, researchers apply a strategy using sinusoidal function for which model can translate long sentences of the training set.

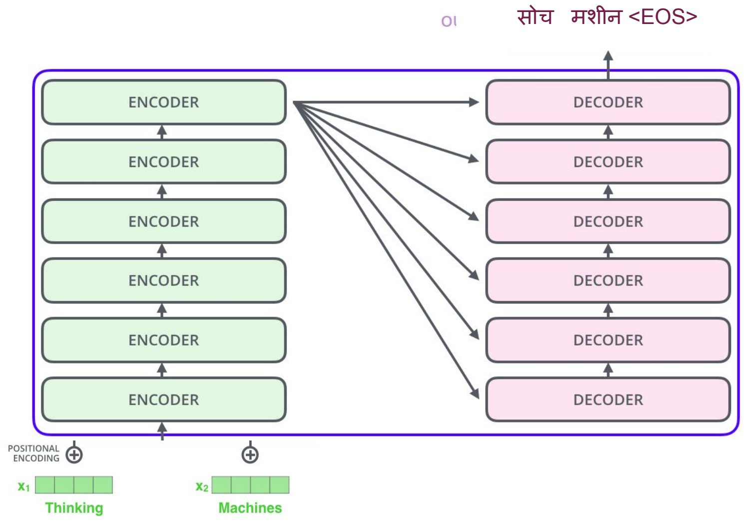


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Encoder

1. Self attention-
   a. For each input token *X*, self attention mechanism generates an output vector *Z* of same size.
   b. Multi-head attention-
      i. Input vectors are processed for multiple sets (heads) to get an output for each set.
      ii. Outputs are combined and processed then to get final encoder output *Z*.
2. Add and normalise - *LayerNorm(X+Z)*
3. Feedforward
4. Add and normalise

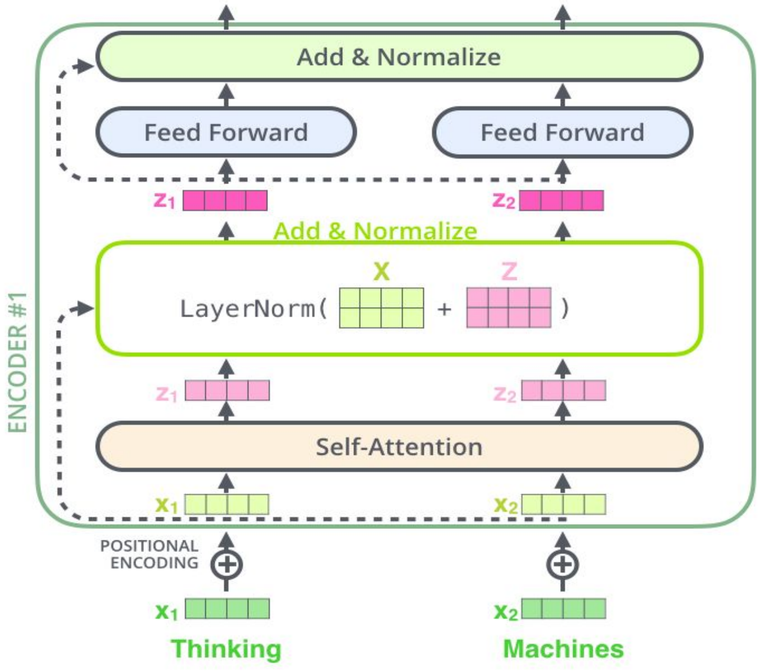
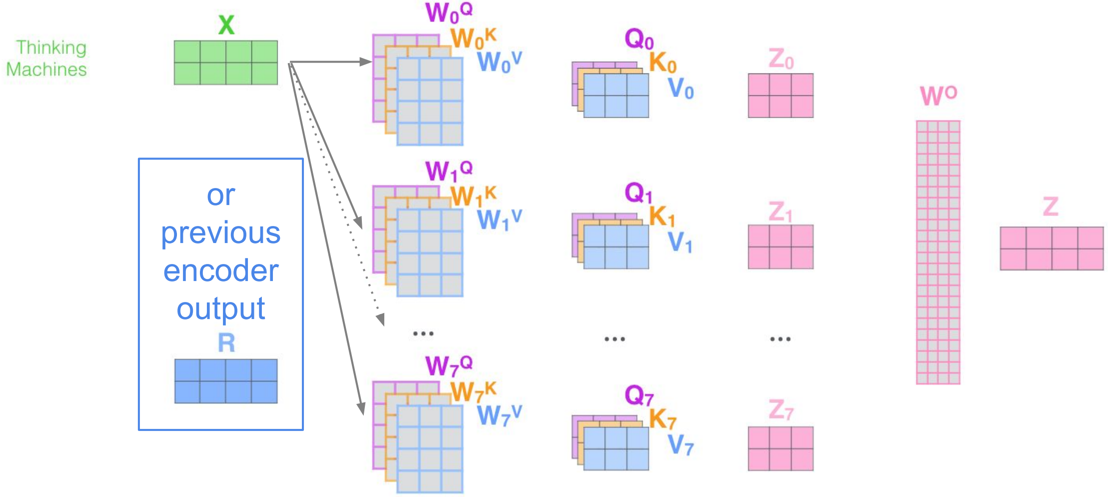


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Multi-head attention



Q: Query; K: Key; V: Value.

1. Multiply the input X (or output R of last encoder) with trainable W_i^Q, W_i^K, W_i^V to get Q_i, K_i, V_i, for each head i . (8 number of heads used).
2. Prepare Z_i of X for i-th head as $\sum softmax((Q.K^T)/\sqrt{d})V^x$, where d is size of Q.
3. Z_i to Z conversion → concatenate then multiply with trainable W^o to transform into a vector Z matching size of X.

Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Enc-Dec attention

Same as self-attention, except–

- **It takes the *K* and *V* from the output of the encoder stack and creates its *Q* from the layer below it.**

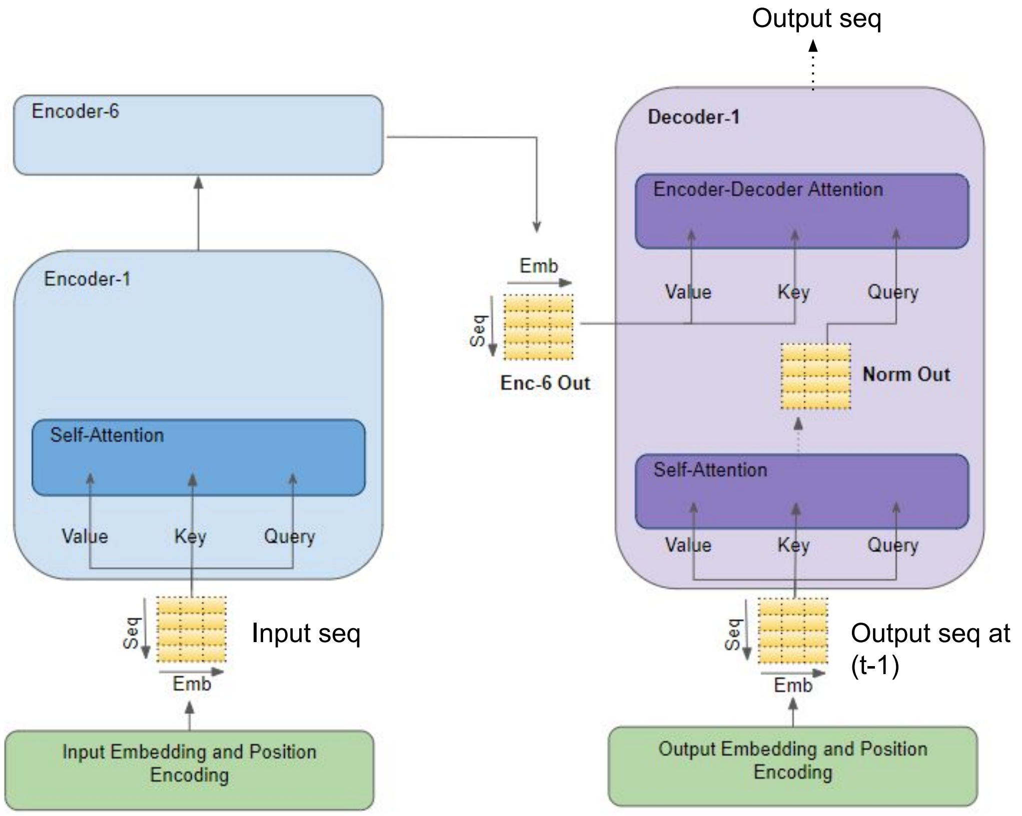


Image source: [https://towardsdatascience.com]

Transformer: Decoder

1. Self-attention: In decoder side the self-attention layer is **only allowed to attend to earlier positions in the output sequence**. This is done by masking future positions.
2. Add and normalize
3. Encoder-decoder attention
4. Add and normalize
5. Feedforward
6. Add and normalize

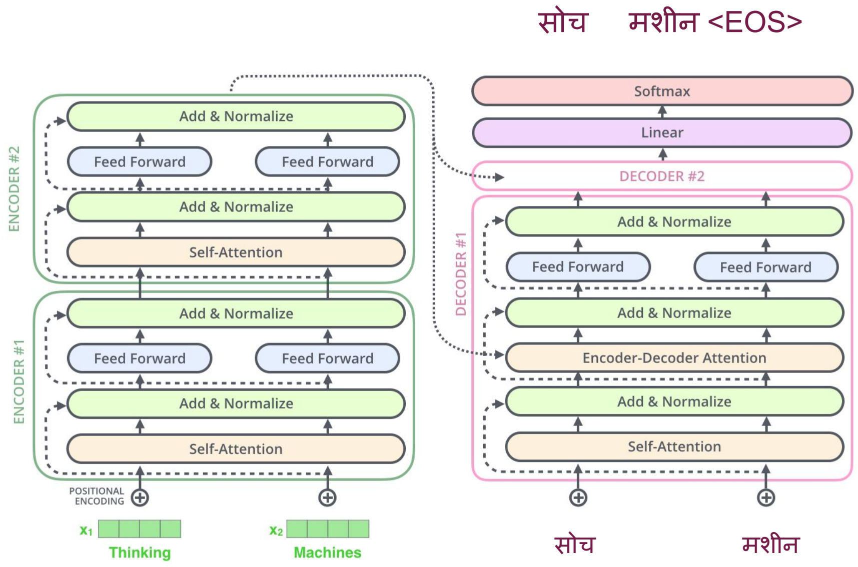


Image source: [jalammar.github.io/illustrated-transformer/]

Transformer: Entire scenario

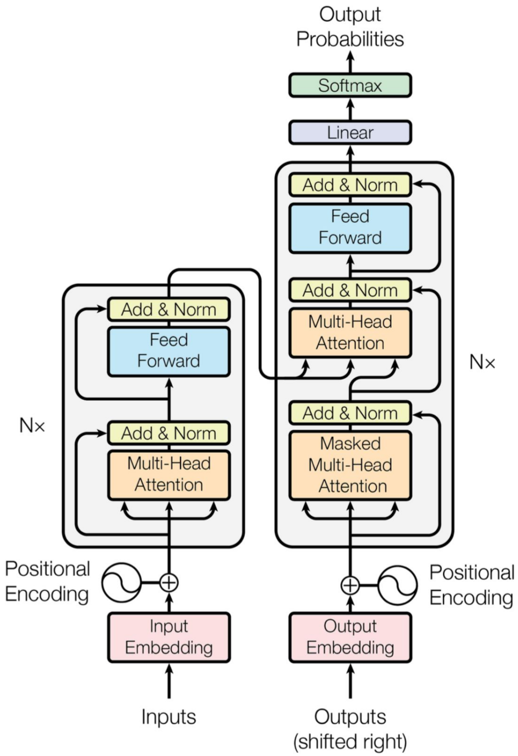


Image source: [11]

Some BLEU scores for Indian Language NMT

Language pair (src\tgt)	Hi	Pa	Bn	Gu	Mr
Hi	-	60.77	28.75	52.17	31.66
Pa	64.67	-	25.32	44.74	27.78
Bn	31.79	26.96	-	24.82	16.61
Gu	55.02	46.48	25.33	-	25.62
Mr	42.97	37.08	21.82	33.29	-

Source: [Dewangan, Shubham, et al. "Experience of neural machine translation between Indian languages." Machine Translation 35.1 (2021): 71-99.]

Summary

- Machine translation is a hard problem because of language divergence.
- BLEU is an automatic evaluation metric to measure the quality of MT output.
- RBMT, EBMT, SMT, and NMT are 4 paradigms of MT.
- From RBMT to NMT, need for human effort decreases with the cost of data availability.
- We discussed RNN-based and Transformer-based encoder-decoder architectures.

LaBSE Filtering

Introduction

Techniques to extract good quality parallel data from the Hindi-Marathi Samanantar Corpus to improve the quality of our Hindi-Marathi MT models.

Motivation

- Neural Machine Translation (NMT) models are *“data hungry”.*
- The comparable corpora have increased tremendously on the World Wide Web, making it an important source for MT task.
- The mined sentence pairs are high in quantity but their quality varies a lot. This affects the quality of the MT systems.
- Hence, there is a need to come up with a preprocessing step to extract only the good quality sentence pairs from the comparable and parallel corpora before passing them to the MT model.

Literature

- Techniques :
  - LaBSE
  - Distilled PML

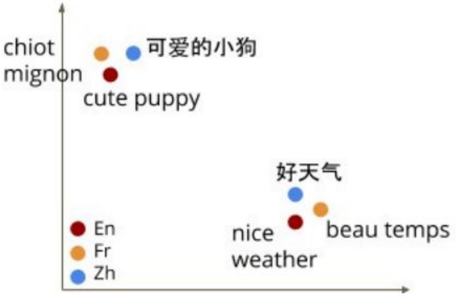
LaBSE
by Google AI

- Language agnostic BERT sentence embedding model is based on a multilingual BERT model.

- Supports 109 languages including some Indic-languages.

LaBSE

- What is Multilingual Embedding Model?
  - that maps text from multiple languages to a shared vector space.
  - Means similar words will be closer and unrelated words will be distant in the vector space as shown in fig:

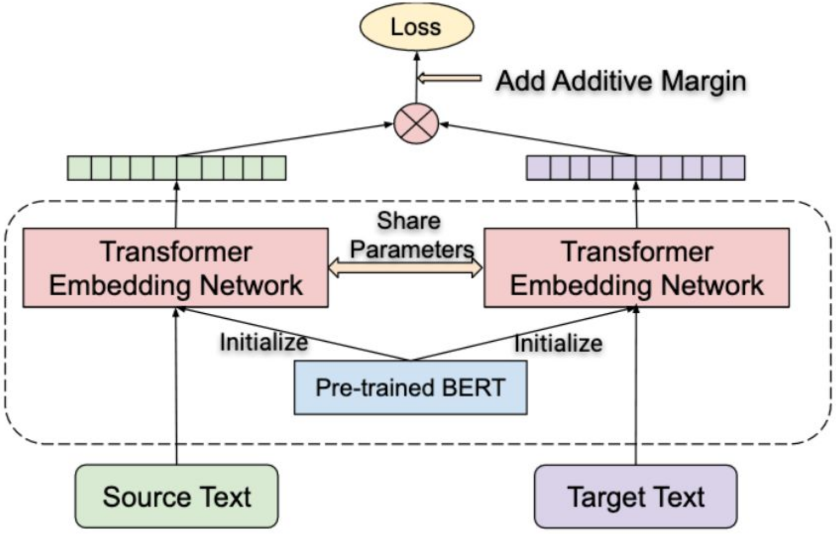


Multilingual Embedding Space via Google AI Blog

Image source- Language agnostic Bert Sentence Embedding [1]

Model Architecture

- The model architecture is based on Bi-Directional dual encoder with an additive margin loss.



Bidirectional Dual Encoder with Additive Margin Softmax and Shared Parameters via LaBSE Paper

Image source- Language agnostic Bert Sentence Embedding [1]

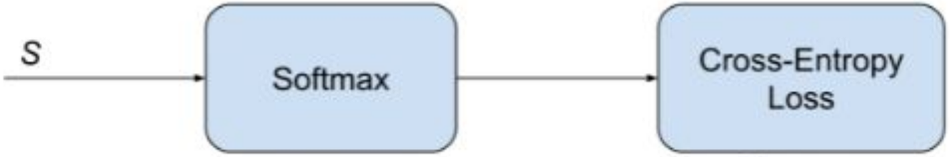
LaBSE Training PIPELINE

- Firstly multilingual BERT model is trained on 109 languages for MLM (Masked Language Modelling) task.

- The obtained BERT encoders is used in parallel at source and target for fine-tuning the Translation Ranking Task.

What is Softmax Loss?

- Confusion? Softmax activation and Softmax loss are different?
- It is a softmax activation followed a Cross-Entropy loss
- It is used for multiclass classification.
- Also known as Categorical Cross-Entropy loss.



$$f(s)_i = \frac{e^{s_i}}{\sum_j^C e^{s_j}} \quad CE = -\sum_i^C t_i log(f(s)_i)$$

Additive Margin Softmax loss

- Motivation :
  - In a classification task, we face a problem when output lies near the decision boundary in the vector space.
  - AM-Softmax aims to solve this by adding a margin to the decision boundary in order to increase the separability of the classes and also making the intra-class distance more compact.

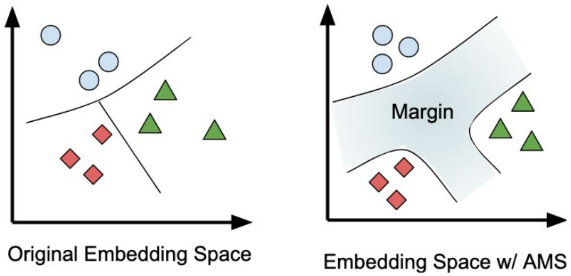


Image source- LaBSE [1]

Literature

- Techniques :
  - LaBSE
  - Distilled PML

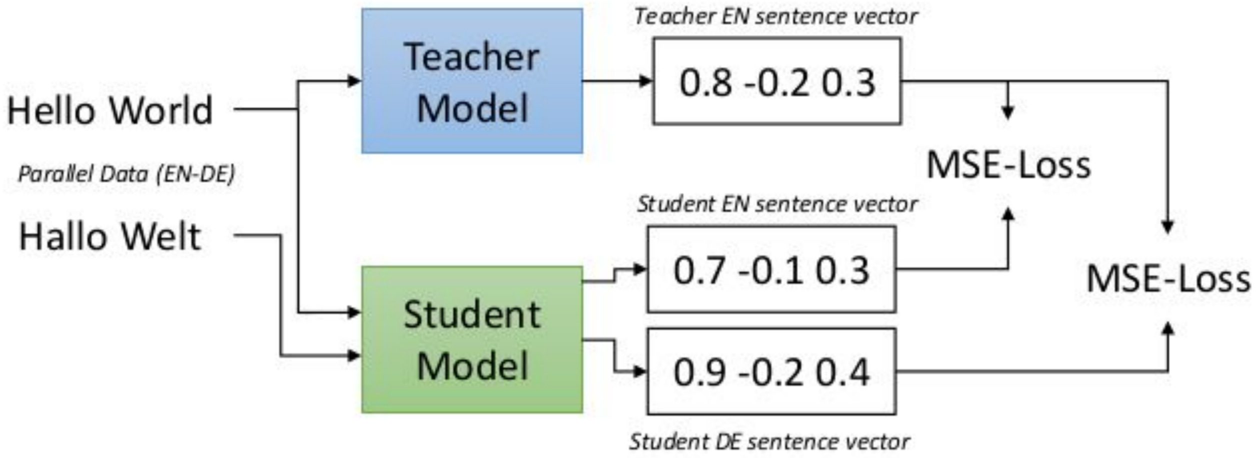
Distilled PML

- Distilled Paraphrase Multilingual Model is a Sentence BERT (SBERT) model extended to multiple languages using multilingual knowledge distillation.

- Knowledge Distillation : Compressing a model by teaching a smaller network exactly what to do at each step using an already bigger trained model.

- A Teacher-Student Model Architecture is use to train Distilled PML model.

Model Architecture



Given parallel data (e.g. English and German), train the student model such that the produced vectors for the English and German sentences are close to the teacher English sentence vector.

Image source- Making Monolingual Sentence Embedding Multilingual using Knowledge Distillation [2]

Model Training

- The English SBERT model is chosen as a teacher model.
- XLM-RoBERTa (XLM-R) model is chosen as a student model.
- So in short student model is trained using XLM-R and further fined tuned on English NLI (Natural language Inference) and STS (Semantic Text Similarity) task using English SBERT model.

Dataset Used (1/3)

- Samanantar Corpus
  - It is the biggest parallel corpus publically available for Indic languages. In our experiments we used Hindi-Marathi Samantar corpus

Dataset	# of Parallel Sentences
Samanantar Corpus	19L

Dataset Used (2/3)

- Combined Corpus:

Dataset	# of Parallel Sentences
PIB	1,08,063
PMI	29,973
Tatoeba	46,277
ILCI	4,62,777
Total Combined Corpus	6,07,832

Dataset Used (3/3)

- Test Datasets:

Corpus Name	# of Test Sentences
WAT21	2390
ILCI	2000

Approach

- LaBSE model is used to generate the sentence embeddings of the Hindi-Marathi Samanantar Corpus.
- These embeddings are used to compute the cosine similarity between the Hindi-Marathi sentence pairs.
- Based on these similarity scores we extract the good quality sentence pairs using a threshold similarity score.
- Then we use these good quality sentence pairs to train the Hindi-Marathi MT systems.

Implementation

- Experiments:
  - Baseline
  - Without LaBSE Filtering
  - LaBSE

Baseline

- We use only the combined corpus to train the Hindi-Marathi Baseline models.
- The combined corpus consists of 6L sentences. The train and tune split given below

Corpus	#Train	#Tune
Combined Corpus + Tatoeba (ILCI + PMI + PIB +Bible +Tatoeba)	6,07,832	14,390

Without LaBSE Filtering

- In this experiment we trained another Hindi-Marathi MT model using the Combined Corpus and whole Samanantar Corpus

- The train and tune split is shown below:

Corpus	# Train	# Tune
Combined Corpus + Tatoeba (ILCI + PMI + PIB +Bible +Tatoeba)	6,07,832	14,390
Samanantar	19,72,689	-
Total	25,80,677	14,390

LaBSE based Filtering

- Hindi-Marathi MT model is trained using the Combined Corpus and LaBSE filtered Samanantar Corpus.

- We use the LaBSE model provided by the huggingface to generate the LaBSE scores for the whole Samanantar Corpus.

- We also computed the LaBSE scores on the PMI corpus, which is a good quality Hindi-Marathi parallel corpus.

- We computed the average LaBSE score which turned out to be 0.89. So we chose 0.9 as the threshold LaBSE score.

LaBSE model provided by the huggingface :
https://huggingface.co/sentence-transformers/LaBSE

Samanantar LaBSE Data Analysis

LaBSE score Range	No. of Parallel Sentences
>=0.9	3,54,315
>=0.91	2,89,802
>=0.92	2,32,187
>=0.93	1,80,776
>=0.94	1,36,200
>=0.95	97,860
>=0.96	65,167
>=0.97	38,699
>=0.98	17,796
>=0.99	4,103

LaBSE based Filtering

- We extracted 3.5L sentences from Samanantar Corpus that had a LaBSE score of 0.9 and above.
- The train, tune split for this model is given below

Corpus	#Train	#Tune + Test
Combined Corpus + Tatoeba (ILCI + PMI + PIB +Bible +Tatoeba)	6,07,832	14,390
Samanantar_labse (labse>=0.9)	3,54,314	-
Total	9,62,146	14,390

Implementation (1/2)

- Training
  - We have used transformer architecture for all our models.
  - We trained the NMT model with the help of OpenNMT-py library

OpenNMT-py is an open source Toolkit
https://github.com/OpenNMT/OpenNMT-py

Implementation (2/2)

- Training
  - The parameters for the transformer model are shown below

Encoder Type	Transformer
Decoder Type	Transformer
Number of layers in encoder/decoder	6
Number of attention heads	8
Size of encoder embedding dimensions	512
Dropout	0.1

Results (1/5)

- Hindi-Marathi MT model

Models	BLEU Score	
	WAT21	ILCI
Baseline	13.8	33.2
Without LaBSE filtering	16.9	33.0
LaBSE filtering	17.8	33.2

We used sacrebleu python library to calculate the BLEU scores.
https://github.com/mjpost/sacrebleu

Results (2/5)

- Hindi-Marathi MT Model
  - We see an increment of 4 BLEU score points in LaBSE filtered model as compared to Baseline on WAT21 test data.
  - Increment of 1 BLEU score points as compared to the “without LaBSE filtered model” on WAT21 test data.
  - We also see that the BLEU score on ILCI dataset remains the same for Baseline and LaBSE filtered model, while it decreases by 0.2 points for “without LaBSE filtered model”.

Results (3/5)

- Marathi-Hindi MT model

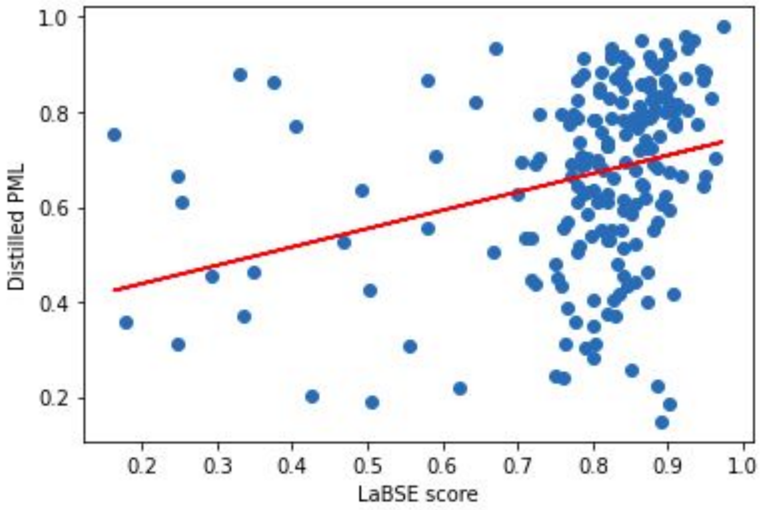
Models	BLEU Score	
	WAT21	ILCI
Baseline	22.1	37.4
Without LaBSE filtering	21.6	33.6
LaBSE filtering	25.1	37.9

Results (4/5)

- Marathi-Hindi MT Model
  - Increment of 3 BLEU score points in LaBSE filtered model as compared to Baseline on WAT21 test data.
  - Increment of 4 BLEU score points as compared to “without LaBSE filtered model” on WAT21 test data.
  - We also see that the BLEU score on ILCI dataset, increments by 0.5 for LaBSE filtered model as compared to Baseline, while it decreases by 4 points for “without LaBSE filtered model”.
  - This is because the Samanantar corpus doesn’t consist of the in-domain data of ILCI dataset.

Results (5/5)

- We also computed the Spearman's rank correlation coefficient between LaBSE and Distilled PML scores.
- These scores were computed on a set of 5000 Hindi-Marathi parallel sentences.
- The correlation coefficient turned out to be **0.38**



Scatter plot of LaBSE and Distilled PML scores

Summary

- Neural Machine Translation systems are “data-hungry”.
- But the quality of the parallel data is as important as the quantity.
- We presented an approach to extract good quality parallel data from the Samanantar Corpus using the LaBSE score to improve the Hi-Mr MT systems.
- This helped us defeat the Baseline Hi-Mr MT systems.

References

[1] Fangxiaoyu Feng and Yinfei Yang and Daniel Cer and Naveen Arivazhagan and Wei Wang. 2020. Language-agnostic BERT Sentence Embedding.

[2] Reimers, Nils and Gurevych, Iryna. 2020. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation.

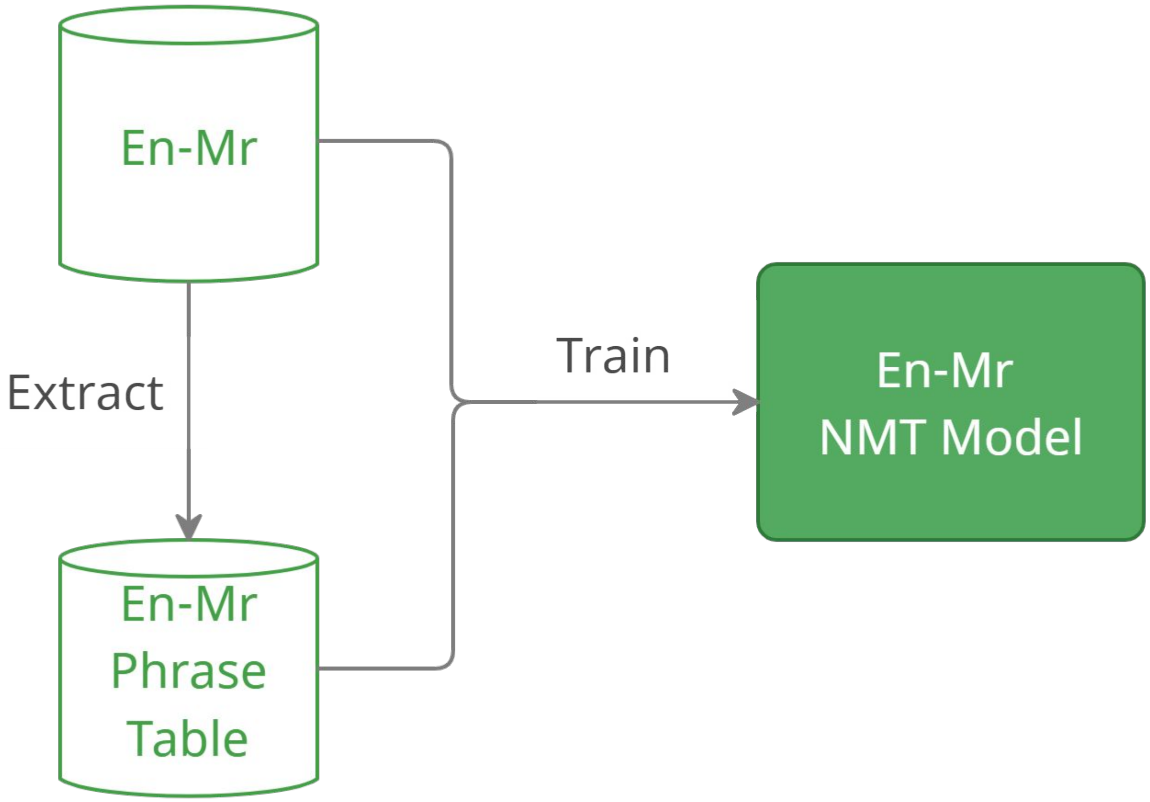
Latest Developments

Phrase Table Injection (1/3)

- In this technique, phrase table is extracted from the Source-Target parallel corpus.

- Finally the Source-Target NMT model is trained using the Source-Target Parallel Corpus and Source-Target Phrases.

Phrase Table Injection (2/3)



Phrase Table Injection (3/3)

- Dataset

	Train	Test (WAT 2021)
Number of Sentences	250,347	2390

- Results

	BLEU Score
Baseline	16.26
Phrase Table Injection	17.15

[12]

Pivoting

- *Pivoting* means utilizing the resources of a related high-resource language for the task of translation between language pairs involving low resource language.

- Example:
  - Utilizing the resources of Hindi for the task of translation between English-Marathi.
  - Utilize the resources of English for the task of translation between Hindi-Marathi or distant language pairs like Russian-Marathi.

Pivoting: Cascade Models

- Cascade Model

Pivoting: Cascade Models

Advantages

- Translation between language pairs that don’t have sufficient parallel corpus, but each of those language has sufficient parallel corpus with pivot language (for example English).

Pivoting: Cascade Models

Disadvantages

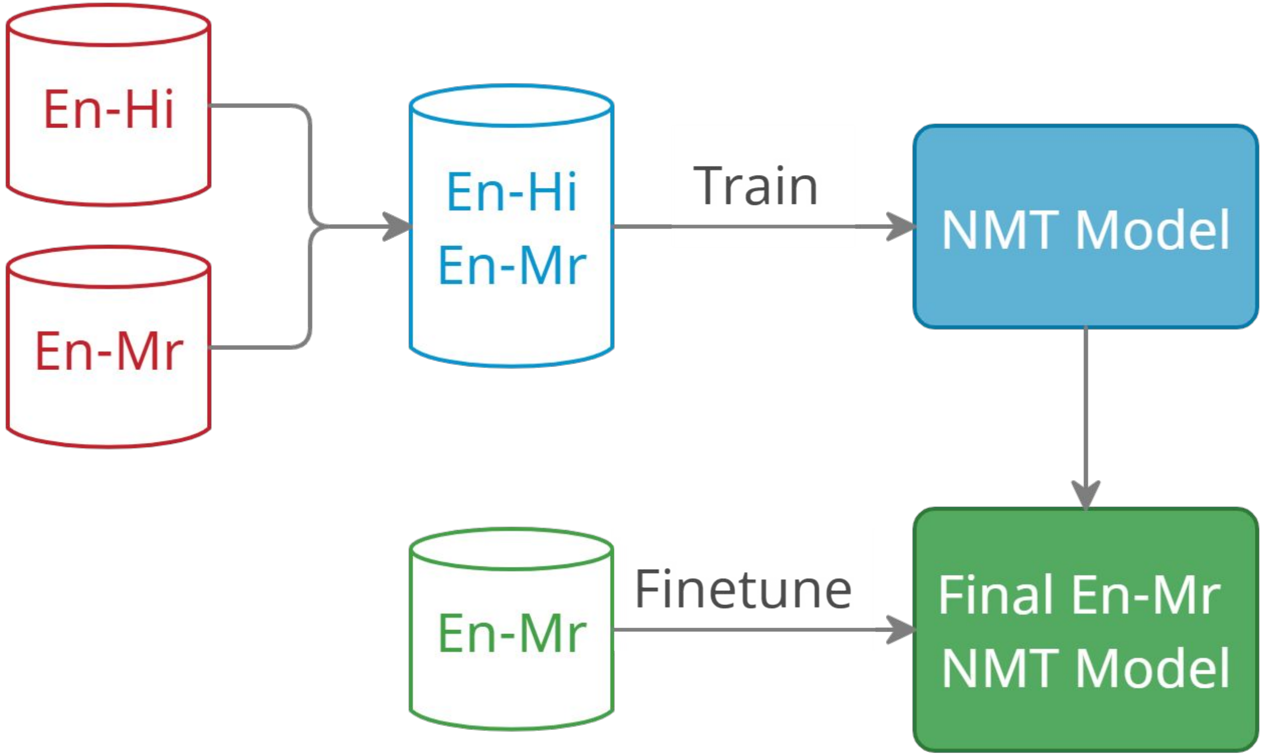
- Double Decoding Time
  - As the source sentence is passed through ***two*** NMT models the decoding time is doubled.

- Propagating Errors
  - As the source sentence is passed through ***two*** NMT models, each model introduces its own errors in translation.

Pivoting: Combined Corpus Model

- In this technique, first the Source-Target and Source-Pivot parallel corpus are combined and a NMT model is trained on this combined data.
- This model is then used as an initialization and the final Source-Target NMT model is trained by finetuning on the Source-Target Parallel corpus.

Pivoting: Combined Corpus Model



Pivoting: Combined Corpus Model

- Initially training the model on combined (En-Hi, En-Mr) the model learned some representation and knowledge from the pivot language data (En-Hi).
- This representation and knowledge can be useful for the final task of En-Mr translation.

Pivoting: Combined Corpus Model

- ## Dataset

	Train	Test (WAT 2021)
Number of Sentences	250,347	2390

- ## Results

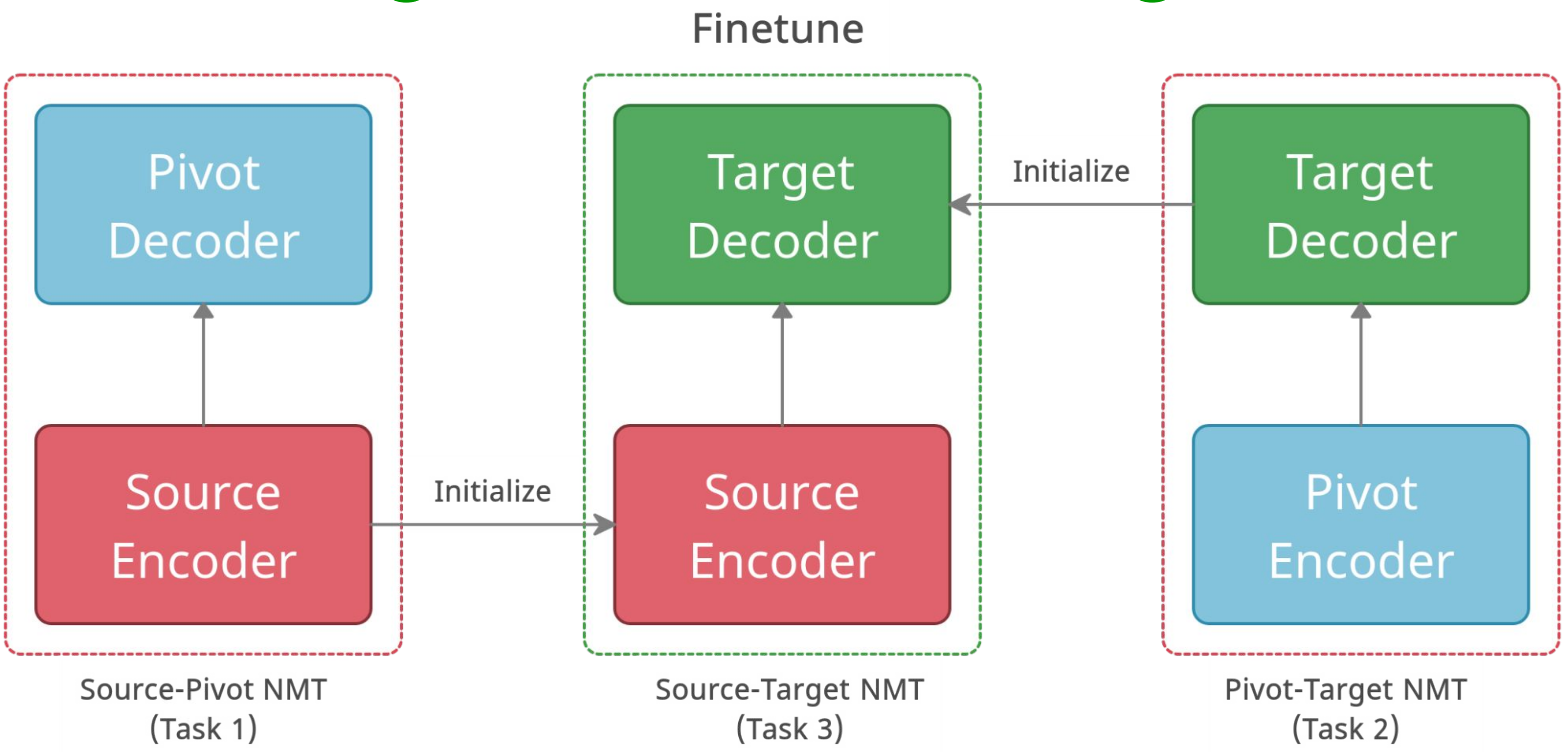
	BLEU Score
Baseline	16.26
Combined Corpus	18.02

[12]

Pivoting: Direct Pivoting Model

- In Direct Pivoting technique we initially train 2 models: source-to-pivot and pivot-to-target.
- Then we initialize the encoder and decoder of the source-to-target model using the encoder of source-to-pivot model and decoder of pivot-to-target model.
- Then we finetune the source-to-target model on source-target parallel data.

Pivoting: Direct Pivoting Model



[13]

Pivoting: Direct Pivoting Model

- Dataset

	Train	Test (WAT 2021)
Number of Sentences	250,347	2390

- Results

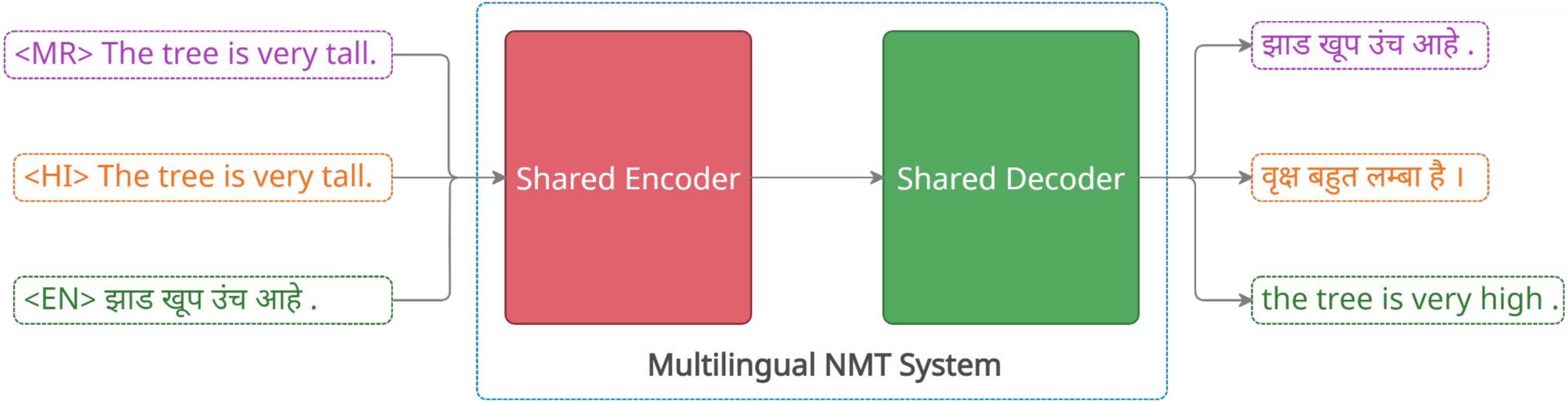
	BLEU Score
Baseline	16.26
Direct Pivoting	16.68

[12]

Multilingual NMT

- Motivation
  - Translation between ***N*** languages to ***N*** languages will require **O(N^2)** models.
  - A single **N-to-N** multilingual model can translate between all **O(N^2)** language directions.
  - Multilingual Models share knowledge between all languages improving performance for low resource language pairs.

Multilingual NMT



- Parameter sharing: Shared encoder and decoder
- Need to find the right amount of shared parameters

Johnson, Melvin, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat et al. "Google's multilingual neural machine translation system: Enabling zero-shot translation." *Transactions of the Association for Computational Linguistics* 5 (2017): 339-351.

Google's MNMT System (Johnson et al., 2017)

Table 1: Many to One: BLEU scores on for single language pair and multilingual models. ⋆: no oversampling

Model	Single	Multi	Diff
WMT De→En	30.43	30.59	+0.16
WMT Fr→En	35.50	35.73	+0.23
WMT De→En⋆	30.43	30.54	+0.11
WMT Fr→En⋆	35.50	36.77	+1.27
Prod Ja→En	23.41	23.87	+0.46
Prod Ko→En	25.42	25.47	+0.05
Prod Es→En	38.00	38.73	+0.73
Prod Pt→En	44.40	45.19	+0.79

Table 2: One to Many: BLEU scores for single language pair and multilingual models. ⋆: no oversampling

Model	Single	Multi	Diff
WMT En→De	24.67	24.97	+0.30
WMT En→Fr	38.95	36.84	-2.11
WMT En→De⋆	24.67	22.61	-2.06
WMT En→Fr⋆	38.95	38.16	-0.79
Prod En→Ja	23.66	23.73	+0.07
Prod En→Ko	19.75	19.58	-0.17
Prod En→Es	34.50	35.40	+0.90
Prod En→Pt	38.40	38.63	+0.23

- A single multilingual model

Massively Multilingual Neural Machine Translation

- Trained Multilingual model on 59 languages. (Aharoni et al., 2019)

	En-Az	En-Be	En-Gl	En-Sk	Avg.
# of examples	5.9k	4.5k	10k	61k	20.3k
baselines	2.16	2.47	3.26	5.8	3.42
one-to-many	**5.06**	**10.72**	**26.59**	**24.52**	**16.72**
many-to-many	3.9	7.24	23.78	21.83	14.19

	En-Ar	En-De	En-He	En-It	Avg.
# of examples	213k	167k	211k	203k	198.5k
baselines	12.95	23.31	23.66	30.33	22.56
one-to-many	**16.67**	**30.54**	**27.62**	**35.89**	**27.68**
many-to-many	14.25	27.95	24.16	33.26	24.9

Table 3: En→X test BLEU on the TED Talks corpus

Unsupervised MT

Unsupervised MT

- No parallel corpus
- Train using only monolingual data
- However, the requirement is:
  - Large monolingual corpus
  - Cross-lingual Word Embeddings

Cross-lingual Word Embeddings

- The geometric relations that hold between words are similar across languages.
  - For instance, numbers and animals in English show a similar (isomorphic) geometric structure as their Spanish counterparts.
- The vector space of a source languages can be transformed to the vector space of the target language t by learning a linear projection with a transformation matrix $W_{s \rightarrow t}$.

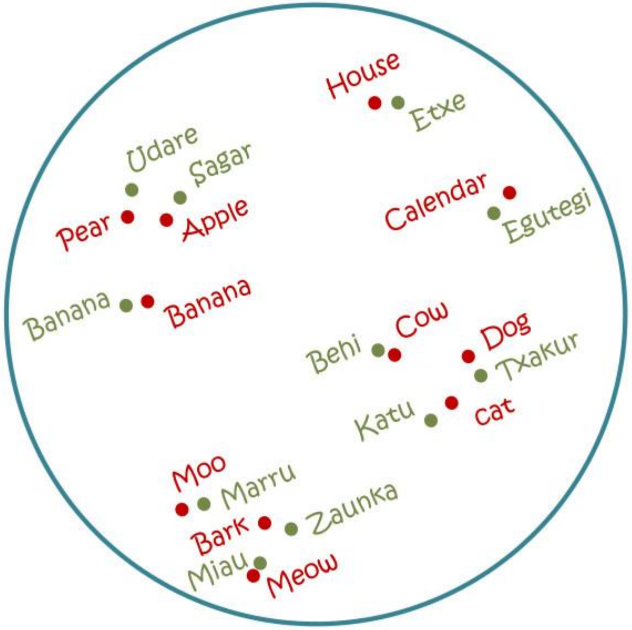
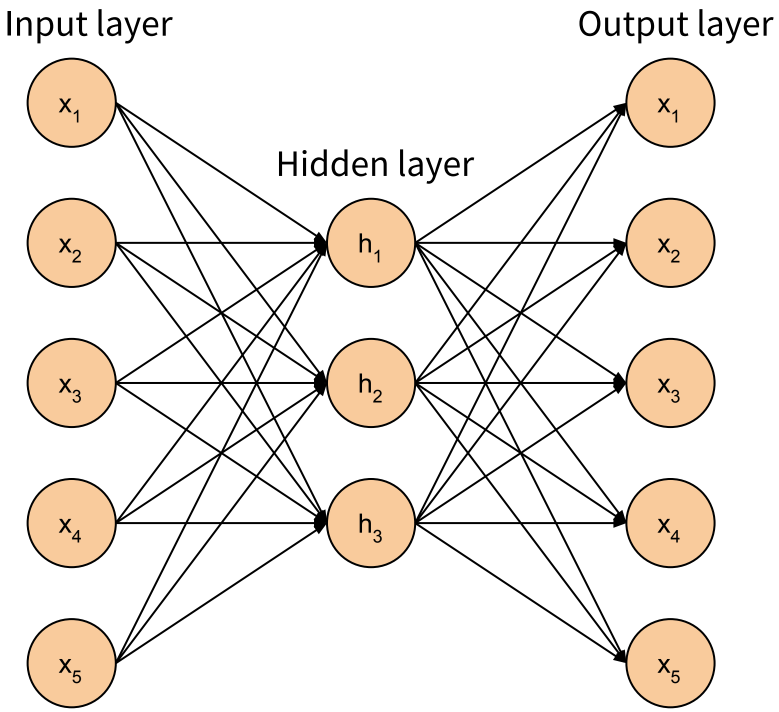


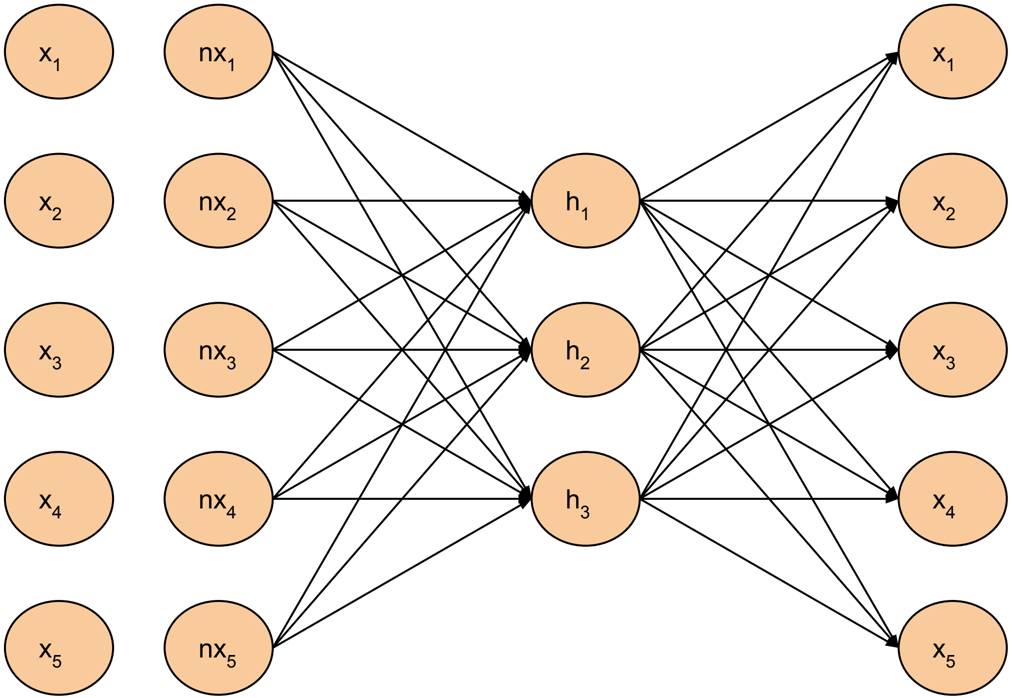
Image source- www.mikelartetxe.com

Autoencoder



- Representation learning
- Neural network to learn reconstruction of the data
- Optimize **Reconstruction Error**
- Balance between
  - Accurately build a reconstruction
  - Handle inputs such that the model doesn't learn to copy the data

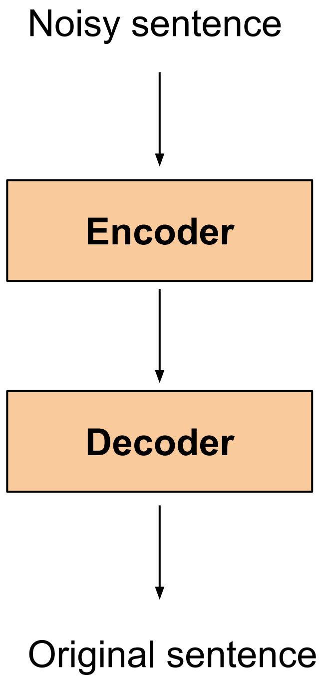
Denoising auto-encoder



Corrupted data

- Learn to generate original sentence from a noisy version of it
- Eliminates the learning of identity function

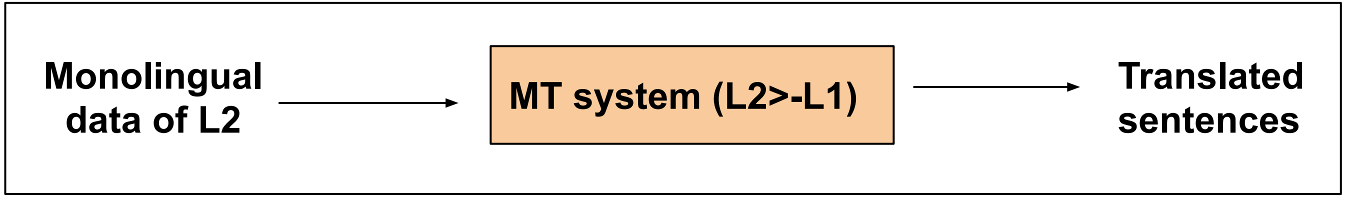
Denoising auto-encoder



- Encoder representation is the representation for noisy sentence
- Decoder tries to generate the original sentence from the encoder representation of the noisy sentence
- A sentence can be corrupted using different types of noise
  - Swapping of words
  - Removal of words
  - Replacement of words with other words

Back-Translation

- Utilize monolingual data of target language
- Generate pseudo parallel data using MT system in opposite direction (target->source)

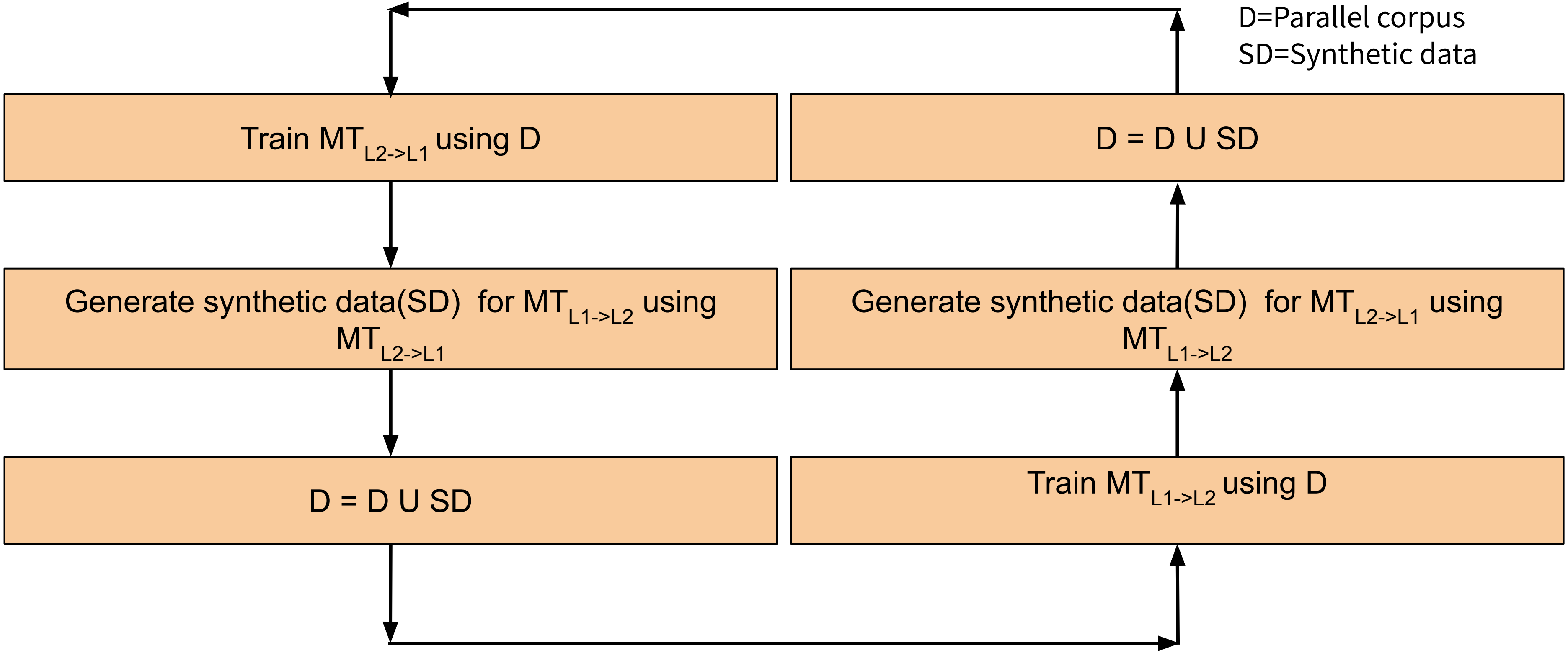


Generation of pseudo parallel sentences

- Train MT system (L1->L2) using a combination of parallel and generated synthetic data both

Sennrich, Rico, Barry Haddow, and Alexandra Birch. "Improving Neural Machine Translation Models with Monolingual Data." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 86-96. 2016.

Iterative Back-Translation



D=Parallel corpus
SD=Synthetic data

Iterative Back-Translation

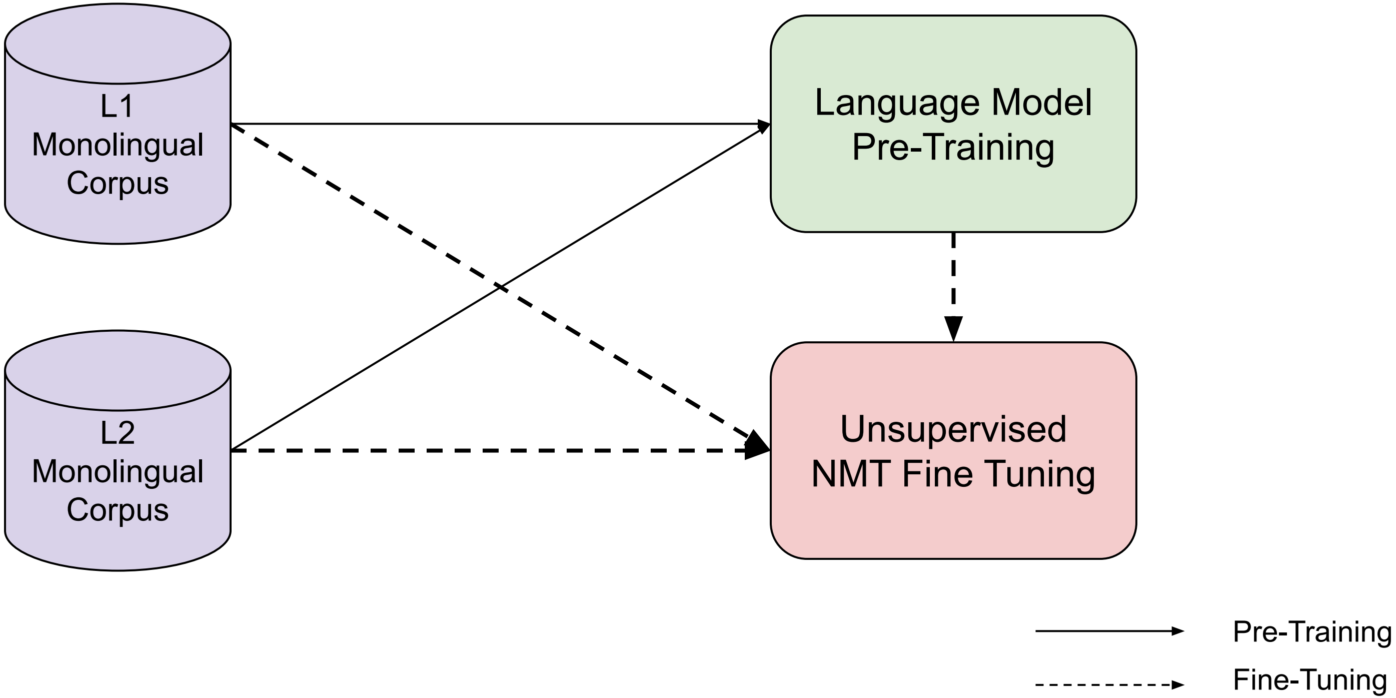
Setting	French–English		English–French		Farsi–English	English-Farsi
	100K	1M	100K	1M	100K	100K
NMT baseline	16.7	24.7	18.0	25.6	21.7	16.4
back-translation	22.1	27.8	21.5	27.0	22.1	16.7
back-translation iterative+1	22.5	-	22.7	-	22.7	17.1
back-translation iterative+2	22.6	-	22.6	-	22.6	17.2

- Beneficial for Low resource languages

Hoang, Vu Cong Duy, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. "Iterative back-translation for neural machine translation." In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pp. 18-24. 2018.

Language model pretraining for Unsupervised NMT

General Framework



mBERT (Devlin et al., 2019)

- Unsupervised pre-training
- Transfer learning with language models
- MLM (Masked language modeling objective) + Next sentence prediction
- Fine-tuned for language understanding and question answering tasks

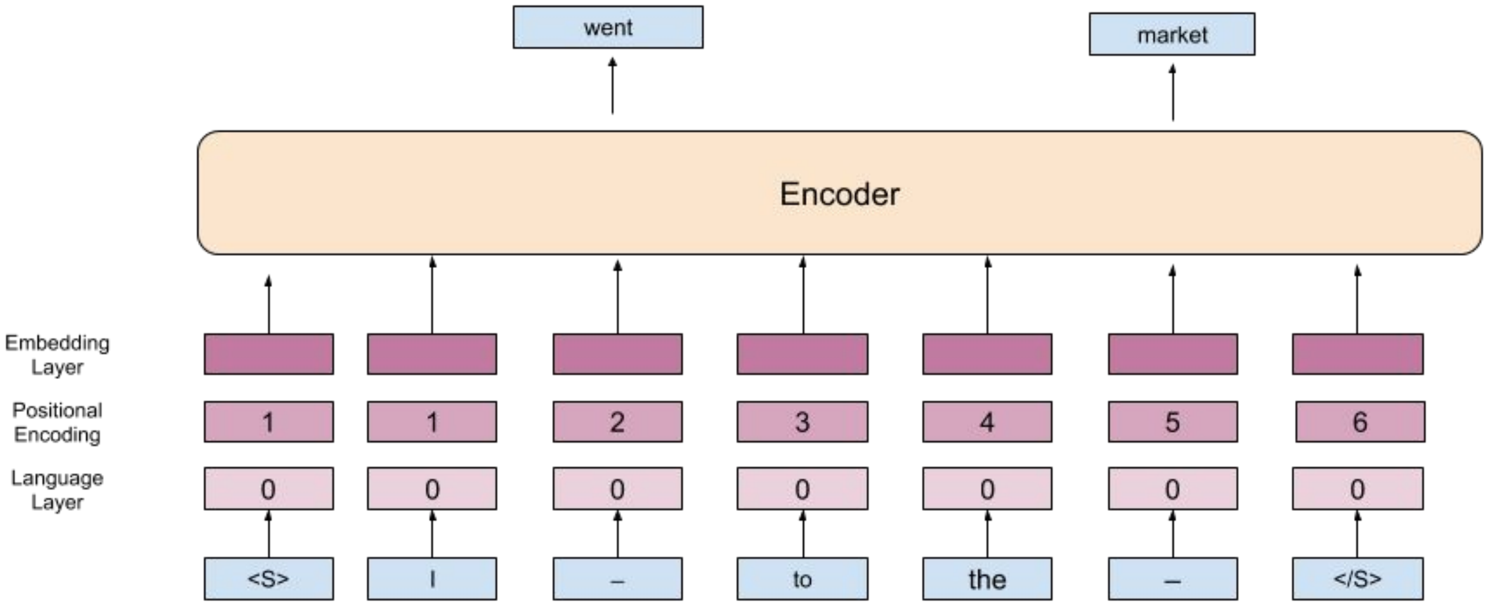
Kenton, Jacob Devlin Ming-Wei Chang, and Lee Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." In *Proceedings of NAACL-HLT*, pp. 4171-4186. 2019.

MuRIL (Khanuja et al., 2021)

- Multilingual LM for Indic languages (16 indic languages and english).
- Transliterated data (Native → Latin)
- MLM (Masked language modeling) + TLM (Translation language modeling) objective
- Fine-tuned for language understanding and question answering tasks

Khanuja, Simran, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam et al. "Muril: Multilingual representations for indian languages." *arXiv preprint arXiv:2103.10730* (2021).

XLM Pre-Training (Conneau et al.,2019)

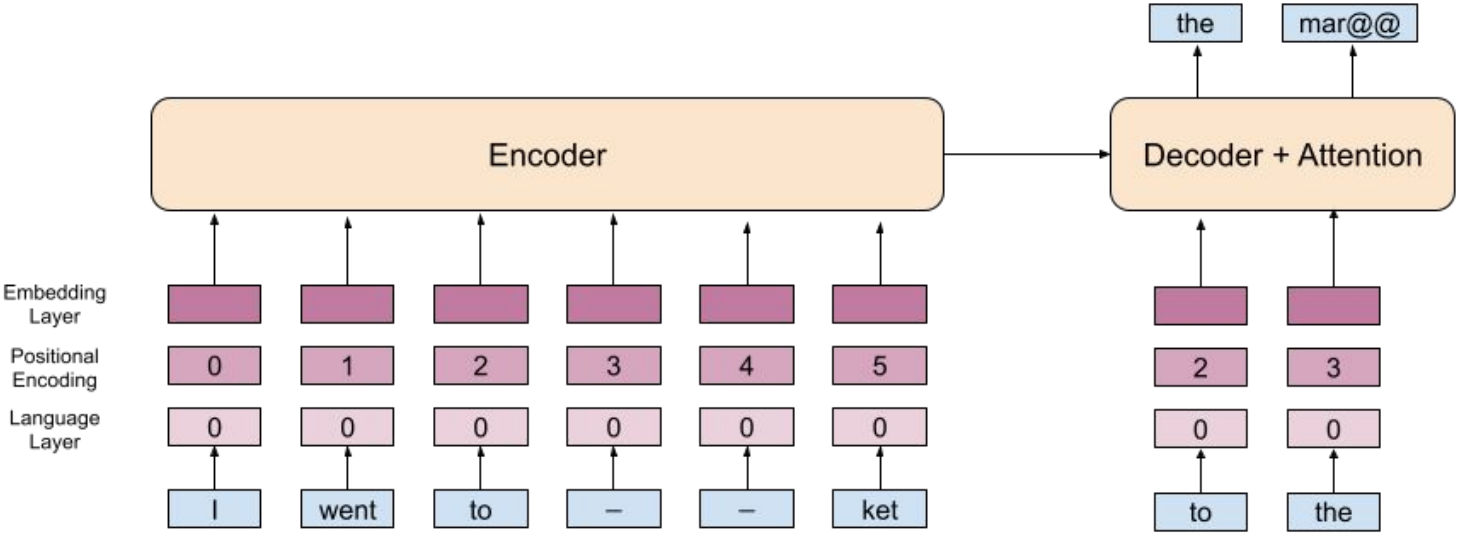


Conneau, Alexis, and Guillaume Lample. "Cross-lingual language model pretraining." *Advances in Neural Information Processing Systems* 32 (2019): 7059-7069.zy

XLM Fine Tuning

- Perform fine-tuning using
  - Iterative back-translation
  - Denoising auto-encoding
- Alternate between the two objective
- Denoising auto-encoding helps in better training of the decoder

MASS Pre-Training (Song et al., 2019)



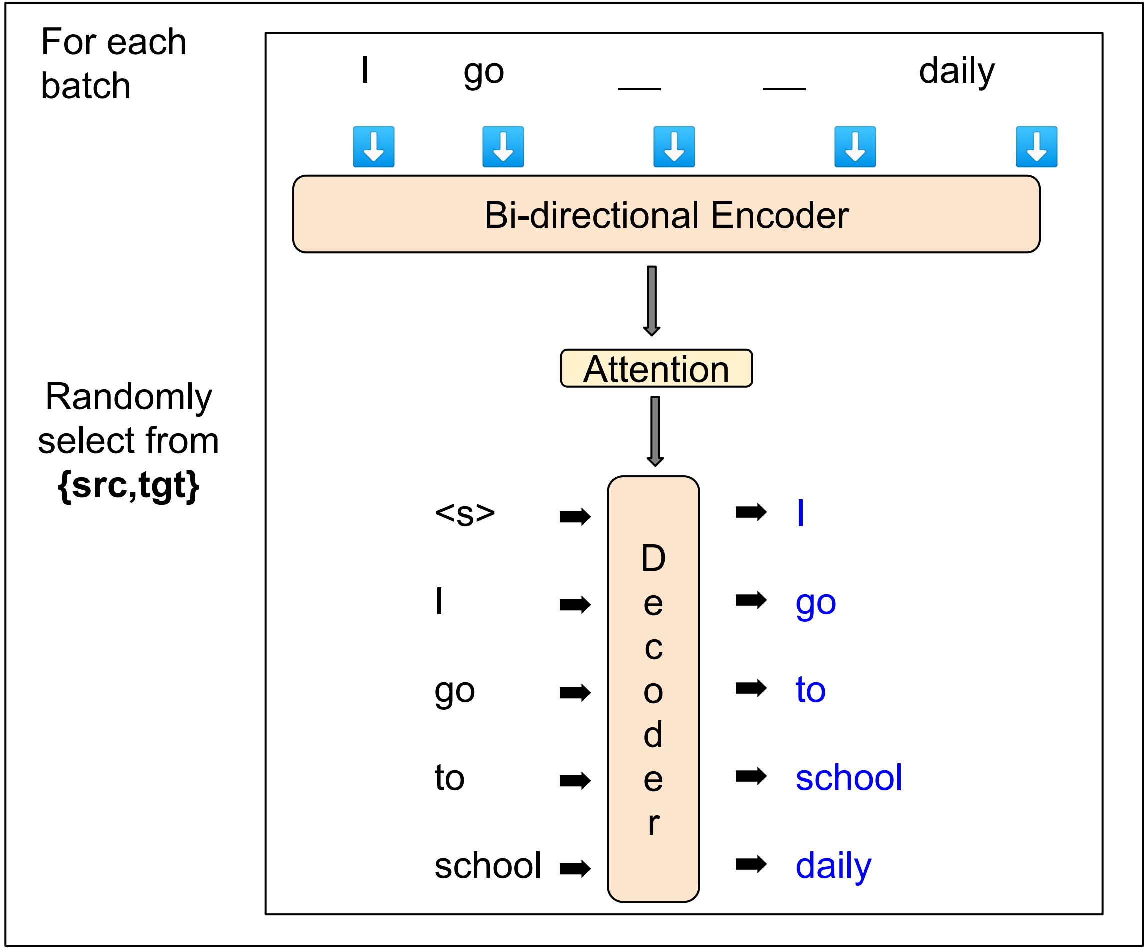
- Perform fine-tuning using : Iterative back-translation

Song, Kaitao, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. "MASS: Masked Sequence to Sequence Pre-training for Language Generation." In *International Conference on Machine Learning*, pp. 5926-5936. PMLR, 2019.

BARTPretraining

- Trained by
  - Corrupting text with an arbitrary noising function
  - Learning a model to reconstruct the original text.
- Denoising full text
- Multi-sentence level

Lewis, Mike, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension." *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, (ACL 2020)*



BART pretraining (possible noising steps) (Lewis et al. 2020)

My name is John. I go to school daily.

Original document

Token Masking	My _ is John. I __ school daily.
Token deletion	My name John. I go to daily.
Text infilling	My _ John. I go _.
Sentence permutation	I go to school daily. My name is John
Document rotation	name is John. I go to school daily. my

mBART (Liu et al 2020)

- A sequence-to-sequence denoising auto-encoder pre-trained on large-scale monolingual corpora in **many languages** using the BART objective
- Unsupervised NMT
  - BART pretraining using monolingual corpora of multiple languages + Iterative Back-Translation

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. arXiv2020.

mBART (Liu et al 2020)

- Pre-training using BART objective on multiple languages

Model	Similar Pairs				Dissimilar Pairs			
	En-De		En-Ro		En-Ne		En-Si	
	←	→	←	→	←	→	←	→
Random	21.0	17.2	19.4	21.2	0.0	0.0	0.0	0.0
XLM (2019)	34.3	26.4	31.8	33.3	0.5	0.1	0.1	0.1
MASS (2019)	**35.2**	28.3	**33.1**	**35.2**	-	-	-	-
mBART	34.0	**29.8**	30.5	35.0	**10.0**	**4.4**	**8.2**	**3.9**

- En-De and En-ro are only trained using specified source and target languages
- En-Ne and En-Si, the pretraining is performed using mBART on 25 languages.
- mBART also generalizes well for the languages not seen in pretraining.

Results: mBART (only on source and target language) pretraining for unsupervised NMT

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. arXiv preprint arXiv:2001.08210, 2020.

Unsupervised NMT for Indic Languages

Motivation

- Unsupervised NMT for Indic languages
- There is lot of cognate overlap between some of the Indic language pairs
- Baseline UNMT (MASS pretraining + Iterative back-translation)
  - low performance for language-pairs with low lexical overlap.

Data

- Monolingual data: AI4Bharat[1]
- Test and validation data: Combination of ILCI[2] and WAT 2020 multi-indic-mt[3] task data (only common sentences are fetched to create validation and test data for indic-indic language pairs)

Language	Number of Sentences
Bengali	7.21 M
Gujarati	7.89 M
Hindi	63.00 M
Malayalam	9.93 M
Marathi	11.70 M
Tamil	21.00 M
Telugu	15.20 M

Language-pair	Size of test data		Size of validation data	
	ILCI	WAT	ILCI	WAT
bn - hi	2000	0	500	382
gu - hi	2000	1403	500	412
ml - hi	2000	869	500	385
mr - hi	2000	1098	500	353
ta - hi	2000	1129	500	371
te - hi	2000	851	500	338

[1] https://indicnlp.ai4bharat.org/corpora/
[2] Jha, Girish Nath. "The TDIL program and the Indian language corpora initiative." In *Language Resources and Evaluation Conference*. 2012.
[3] http://lotus.kuee.kyoto-u.ac.jp/WAT/indic-multilingual/

Approaches

- The lexical divergence between source and target languages play a big role in the success of UNMT.
- We explore following approaches
  - Baseline unsupervised NMT (MASS pretraining + iterative back-translation)
  - **Script conversion** (Transliteration to a common script)
  - **Unsupervised bilingual embedding based initialization** to bring the vocabulary of the two languages closer
  - **Dictionary word substitution** using a bilingual dictionary.
    - Randomly replace whole words in the sentence with the corresponding word translation obtained from a ground truth bilingual dictionary as a preprocessing step

Results: Language-pairs with High Lexical Overlap

	BLEU		CHRF2	
	hi → bn	bn → hi	hi → bn	bn → hi
Original Script	0.94	1.43	0.16	0.17
Bilingual BPE Embeddings	0.62	0.82	0.14	0.15
Anchored Cross-lingual Pre-training	0.38	0.78	0.15	0.16
Code Switching Pre-training	0.35	0.54	0.13	0.14
Dictionary Word Substitution	0.95	1.59	0.17	0.17
Transliteration (T)	5.60	7.53	0.34	0.34
Bilingual BPE Embeddings + T	7.60	10.65	0.37	0.37
Anchored Cross-lingual Pre-training + T	4.78	8.58	0.32	0.34
Code Switching Pre-training + T	2.70	3.46	0.26	0.26
Dictionary Word Substitution + T	**8.34**	**12.16**	**0.39**	**0.37**

	BLEU		CHRF2	
	hi → gu	gu → hi	hi → gu	gu → hi
Original Script	11.36	12.99	0.32	0.31
Bilingual BPE Embeddings	16.5	19.57	0.40	0.40
Anchored Cross-lingual Pre-training	6.92	9.57	0.25	0.28
Code Switching Pre-training	2.81	2.31	0.16	0.18
Dictionary Word Substitution	9.83	11.14	0.29	0.29
Transliteration (T)	21.72	26.01	0.53	0.53
Bilingual BPE Embeddings + T	21.72	26.01	0.56	0.56
Anchored Cross-lingual Pre-training + T	17.77	24.33	0.48	0.51
Code Switching Pre-training + T	12.92	14.54	0.41	0.41
Dictionary Word Substitution + T	**26.38**	**33.02**	**0.57**	**0.58**

	BLEU		CHRF2	
	hi → mr	mr → hi	hi → mr	mr → hi
Original Script	9.49	15.89	0.40	0.41
Bilingual BPE Embeddings	10.85	18.57	0.43	0.44
Anchored Cross-lingual Pre-training	7.86	12.79	0.36	0.39
Code Switching Pre-training	5.91	7.53	0.3	0.31
Dictionary Word Substitution	**12.11**	**20.37**	**0.44**	**0.45**

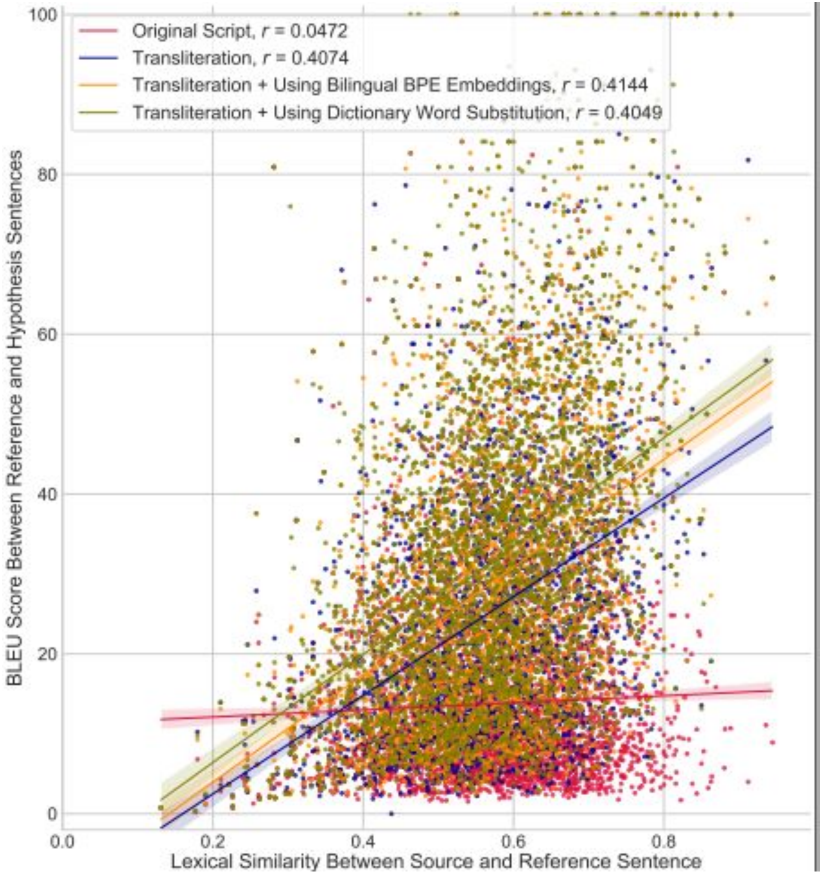
Results: Language-pairs with Low Lexical Overlap

	BLEU		CHRF2	
	hi → ml	ml → hi	hi → ml	ml → hi
Original Script	0.00	0.44	0.16	0.15
Bilingual BPE Embeddings	0.15	0.79	0.18	0.15
Anchored Cross-lingual Pre-training	0.00	0.00	0.31	0.15
Code Switching Pre-training	0.00	0.21	0.14	0.14
Dictionary Word Substitution	0.00	0.39	0.17	0.15
Transliteration (T)	0.46	2.25	0.23	0.22
Bilingual BPE Embeddings + T	0.95	3.66	0.27	0.25
Anchored Cross-lingual Pre-training + T	0.2	1.28	0.17	0.2
Code Switching Pre-training + T	0.17	0.43	0.16	0.17
Dictionary Word Substitution + T	**1.04**	**1.04**	**0.28**	**0.25**

	BLEU		CHRF2	
	hi → ta	ta → hi	hi → ta	ta → hi
Original Script	0.22	0.59	0.21	0.15
Bilingual BPE Embeddings	0.47	0.08	0.14	0.01
Anchored Cross-lingual Pre-training	0.39	0.67	0.15	0.16
Code Switching Pre-training	0	0	0.18	0.13
Dictionary Word Substitution	0	0.39	0.2	0.14
Transliteration (T)	0.43	1.15	0.23	0.18
Bilingual BPE Embeddings + T	0.92	2.46	0.27	0.21
Anchored Cross-lingual Pre-training + T	**1.67**	2.34	0.22	**0.24**
Code Switching Pre-training + T	0.24	0.44	0.19	0.14
Dictionary Word Substitution + T	0.94	**2.58**	**0.27**	0.20

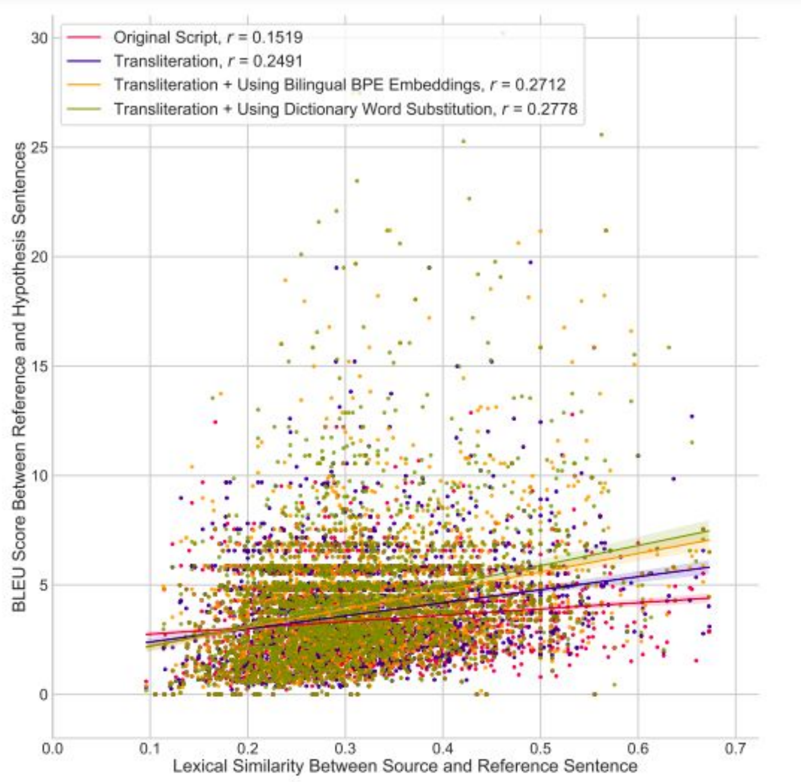
	BLEU		CHRF2	
	hi → te	te → hi	hi → te	te → hi
Original Script	0.49	1.35	0.17	0.17
Bilingual BPE Embeddings	1.19	2.59	0.2	0.19
Anchored Cross-lingual Pre-training	0.38	0.7	0.15	0.16
Code Switching Pre-training	0.21	0.48	0.14	0.15
Dictionary Word Substitution	0.38	0.68	0.15	0.15
Transliteration (T)	1.67	3.45	0.24	0.25
Bilingual BPE Embeddings + T	3.11	6.9	0.30	0.31
Anchored Cross-lingual Pre-training + T	1.71	2.38	0.22	0.24
Code Switching Pre-training + T	0.87	1.43	0.19	0.21
Dictionary Word Substitution + T	**3.38**	**7.8**	**0.31**	**0.32**

Lexical Similarity versus BLEU score



- The correlation between lexical similarity between source and reference sentences and BLEU score for **Gujarati →Hindi**.

Lexical Similarity versus BLEU score



- The correlation between lexical similarity between source and reference sentences and BLEU score for **Hindi → Malayalam**.
- r value with original script is very low.

Shared Vocabulary

Lang-pair (src - tgt)	Original Script		Shared Script	
	% source tokens present in target	% target tokens present in source	% source tokens present in target	% target tokens present in source
bn - hi	4.04	1.21	13.65	4.07
gu - hi	2.86	1.31	19.43	8.88
ml - hi	0.0	0.0	1.1	1.82
mr - hi	14.15	7.11	14.15	7.11
ta - hi	0.0	0.0	0.62	1.09
te - hi	0.45	0.36	3.65	2.87

Statistics on Lexical overlap between the two languages

Qualitative Analysis

- Fluent but inadequate translations

Malayalam	Source (Devanagari) (English meaning)	കഴിക്കാവുന്നത്രയേ വിളമ്പാവൂ कळिक्कावुन्नत्रये विळम्पावू . Take as much as you can eat
Hindi	Reference	हम उतना ही लें , जितना खाना है
	Translation by Original Script (Gloss)	पहलवान ने वेटर को निकाल डाला । wrestler has waiter removed
	Translation by Transliteration (Gloss)	करने जा रहे हैं , ये चिंता बारीकी . to-do go are is these thinking closely
	Translation by Bilingual BPE Embeddings + T (Gloss)	भोजन करने लगे , ये बरतन food to-do these utensils
	Translation by Dictionary Word Substitution + T (Gloss)	खाने के लिए ये फल eating of for these fruits

Qualitative Analysis

- Meaning drift due to transliteration

Bengali	Source (Devanagari) (English meaning)	হেল্দি লাইফ স্টাইল মেনে চলুন। हेल्दि लाइफ स्टाइल मेने चलुन Adopt a healthy life-style
Hindi	Reference	हेल्दी लाइफ स्टाइल अपनाएँ।
	Translation by Original Script (Gloss)	लॉट्सबर्ग फ्रेम केयर शुरू , तहलका lotsburg frame care start, panic
	Translation by Transliteration (Gloss)	हेल्दी लाइफ स्टाइल के लिए चल रहे हैं: healthy life-style for walking are
	Translation by Bilingual BPE Embeddings + T (Gloss)	हेल्दी लाइफ स्टाइल भी चल रही है ... healthy life-style too walk doing is
	Translation by Dictionary Word Substitution + T (Gloss)	हेल्दी लाइफ स्टाइल के लिए चल रहे हैं healthy life style for walking are

- **চলুন** (Chaluna) → to go, মেন চলুন (mene chaluna) →**maintain**
- The Hindi translation for the verb chaluna is **चलो** (chalo). [This happens because of transliteration and bilingual embeddings]

Summary

- Current state of the art approaches in unsupervised NMT
- Analysis of Unsupervised NMT for Indic languages
  - The lexical divergence between source and target language plays an important role.
  - 3 approaches to bridge lexical divergence between source and target languages
    - Script conversion
    - Initialization using bilingual embeddings
    - Dictionary word substitution

Machine Translation Demonstration

Automatic Post Editing

Outline

1. Motivation
2. Problem Statement
3. Challenges
4. Categorization of APE Systems
5. APE Paradigms
6. WMT APE Shared Tasks
7. HW-TSC’s APE System
8. Experiments and Results

Motivation

- Machine Translation (MT) systems : far from perfect
- Requirement of post-processing through human intervention
  - Generation of parallel data (mt_op <--> post-edited mt_op)
- Can we automate the post-processing phase using this data?
- Use cases:
  - Ideal: to eliminate the need of human involvement.
  - Black-box scenario: To further improve translations by identifying and correcting recurring MT errors
  - Adapt terminologies for a specific domain

Problem Statement

- Automatic Post Editing (APE) : Given the translations generated by a machine translation system, generate corrected versions of them which are publishable.
  - The edits should be minimal.
- In a supervised setting, training data contains triplets:
  - Source sentence: People can **get** COVID-19 even after vaccination.
  - MT translation: लसीकरणानंतरही लोकांना कोविड - 19 **मिळू** शकतो .
  - Human post-edited version: लसीकरणानंतरही लोकांना कोविड - 19 **होऊ** शकतो .
- Input: MT translation, Output: Human post-edited version

Categorization of APE systems

- APE Systems can be categorized as follows:
  - Accessibility of MT System: Black-box or Glass-box
  - Type of Post-editing Data: Real or Synthetic
  - Domain of the Data: General or Specific
- We focus on:
  - Black-box scenario
  - Real as well as Synthetic Data
  - Domain Specific APE systems

Challenges (1/2)

- Data :
  - Deep Learning based Methods: data-hungry
  - Data Sparsity
  - Increased complexity: Same error can be corrected in more than one way
  - Coverage of error-correction patterns
  - Requirement of new datasets

Challenges (2/2)

- Technology :
  - Neural APE systems: Follow similar trend as MT
  - Joint modelling of SRC and MT_OP helps, but increases data sparsity
  - Poses a risk of unnecessary edits
  - Issue of overfitting
  - Requirement: Techniques resilient to problem of over-correction, and can work in low-resource settings

APE Paradigms (1/2)

- APE task: a monolingual translation task
  - The same MT technology has been used for APE
- Rule-based APE:
  - Not much work done
  - Uses precise PE rules
  - The rules might not be capturing all possible scenarios
  - Not portable across domains

APE Paradigms (2/2)

- Phrase-based APE:
  - Dominated the APE field for a few years
  - Showed significant improvements when underlying MT system was rule-based
  - Limited improvements when underlying MT system was SMT
- Neural APE:
  - Current-state-of-the-art
  - Showed significant improvements when underlying MT system is SMT

Terminologies

- SRC: source language sentence
- MT_OP: translation of SRC generated using a MT system
- MT_REF: reference target language sentence for the SRC
- PE_REF: Human post-edited version of MT_OP
- PE_OP: Output generated by the APE system
- Triplet: (SRC, MT_OP, PE_REF)
- Example:
  - SRC: People can **get** COVID-19 even after vaccination.
  - MT_OP: लसीकरणानंतरही लोकांना कोविड - 19 **मिळू** शकतो .
  - PE_REF: लसीकरणानंतरही लोकांना कोविड - 19 **होऊ** शकतो .

WMT APE Shared Tasks

WMT APE Shared Task

- WMT APE Shared Task is hosted every year since 2015
- Considers the black-box scenario
- APE data is shared across participants
  - Train, development data: (SRC, MT_OP, PE_REF)
  - Test data: (SRC, MT_OP)
- Participants submit PE_OP for the test set
- The submitted systems are evaluated over the benchmark datasets
- Evaluation metrics used for rankings
  - TER and BLEU

WMT APE Shared Task

- Baseline APE systems:
  - "do nothing" APE system
    - Does not make any modification to MT_OP, i.e. PE_OP = MT_OP
- Participants are allowed to use any external data.
- Optional resources - Synthetic corpora:
  - Artificial Corpus (4 Million triplets) [6]
  - eSCAPE corpus (14.4 Million triplet) [7]

WMT15 APE Shared Task

- Pilot round
- Language pair: English - Spanish
- Domain: News
- Data: Train (12,000), Development (1,000), Test(2,000) triplets
- PE_REF was collected through crowd-sourcing
- Number of submissions: 7
- All the submissions were based on phrase-based technology
- None of the submissions beat the baseline

WMT15 APE Shared Task

- Difference between
  - the baseline and top ranked system: -0.315 TER points
  - the phrase based APE and top ranked system: 0.926
- It was hypothesized that the poor results are due to origin, quantity and domain of the data.

- Results:

ID	Avg. TER
Baseline	22.913
FBK Primary	23.228
LIMSI Primary	23.331
USAAR-SAPE	23.426
LIMSI Contrastive	23.573
Abu-MaTran Primary	23.639
FBK Contrastive	23.649
(Simard et. al) [124]	23.839
Abu-MaTran Contrastive	24.715

Image Source: [1]

WMT16 APE Shared Task

- Language pair: English - German
- Domain: IT domain
- Data: Train (12,000), Development (1,000), Test(2,000) triplets
- PE_REF was collected through professional post-editors
- Number of submissions: 11
- Except two, rest of the submissions were based on phrase-based technology
- 7 teams crossed the baseline.
- The top ranked system used
  - RNN based encoder-decoder model with attention
  - Artificially generated Data of around 4 million triplets (Using Back Translation)

WMT16 APE Shared Task

- Difference between
  - the baseline and top ranked system: 3.24 TER points
  - the phrase based APE and top ranked system: 3.12
- Have the results improved due to change of data or due to technology shift?

- Results:

ID	Avg. TER (↓)	BLEU (↑)
AMU Primary	21.52	67.65
AMU Contrastive	23.06	66.09
FBK Contrastive	23.92	64.75
FBK Primary	23.94	64.75
USAAR Primary	24.14	64.10
USAAR Constrastive	24.14	64.00
CUNI Primary	24.31	63.32
(Simard et al.)[124]	24.64	63.47
Baseline	24.76	62.11
DCU Contrastive	26.79	58.60
JUSAAR Primary	26.92	59.44
JUSAAR Contrastive	26.97	59.18
DCU Primary	28.97	55.19

Image Source: [1]

WMT17 APE Shared Task

- Language pairs and domain:
  - English - German (IT)
  - German - English (Medical)
- Data for English-German: Train (11,000), Test(2,000) triplets
- Data for German-English: Train(25,000), Dev(1,000), Test(2,000)
- Number of submissions: 15 (English-German), 5(German-English)
- Most of the submissions were based on neural approaches, and beat the baseline except one system.
- The top ranked system followed neural APE approach:
  - Used multi-source APE System
  - Trained the system over synthetic corpus and fine-tuned on the provided in-domain data

WMT17 APE Shared Task

- Difference between
  - the baseline and top ranked system: 4.88 TER points
  - the phrase based APE and top ranked system: 5.09 TER points

- Results (English-German):

ID	Avg. TER (↓)	BLEU (↑)
FBK Primary	19.6	70.07
AMU Primary	19.77	69.5
AMU Contrastive	19.83	69.38
DCU Primary	20.11	69.19
DCU Contrastive	20.25	69.33
FBK Contrastive	20.3	69.11
FBK_USAAR Contr.	21.55	67.28
USAAR Primary	23.05	65.01
LIG Primary	23.22	65.12
JXNU Primary	23.31	65.66
LIG Contrastive-Forced	23.51	64.52
LIG Contrastive-Chained	23.66	64.46
CUNI Primary	24.03	64.28
USAAR Contrastive	24.17	63.55
Baseline	24.48	62.49
(Simard et al.)[124]	24.69	62.97
CUNI Contrastive	25.94	61.65

Image Source: [2]

WMT17 APE Shared Task

- Difference between
  - the baseline and top ranked system: 0.26 TER points
  - the phrase based APE and top ranked system: 0.45 TER points
- Difference between the baseline and phrase based APE can be seen as an indicator of task difficulty level.

- Results (German-English):

ID	Avg. TER (↓)	BLEU (↑)
FBK Primary	15.29	79.82
FBK Contrastive	15.31	79.64
LIG Primary	15.53	79.49
Baseline	15.55	79.54
LIG Contrastive-Forced	15.62	79.48
LIG Contrastive-Chained	15.68	79.35
(Simard et al.)[124]	15.74	79.28

Image Source: [2]

WMT18 APE Shared Task

- Language pairs and domain: English - German (IT)
- Two different MT systems: PBMT, NMT
- Data for PBMT: datasets released in earlier rounds
- Data for NMT: Train (13,442), Dev (1,000), Test (1,023)
- Number of submissions: 11 (PBMT), 10 (NMT)
- All the submissions were based on neural approaches
- This allowed to compare the effectiveness of neural APE systems
- The top ranked system:
  - Used multi-source APE System and transformer architecture
  - Trained the system over large synthetic corpus and fine-tuned on the provided in-domain data

WMT18 APE Shared Task

- Difference between the baseline and top ranked system: 6.24 TER points

- Results (PBMT):

ID	TER (pe)	BLEU (pe)
MS_UEdin Primary	18.0	72.52
FBK Contrastive (MRT+MLE)	18.62	71.04
FBK Primary (MRT)	18.94	71.22
POSTECH Contrastive (fix5)	19.63	69.87
POSTECH Primary	19.72	69.8
POSTECH Contrastive (var5)	19.74	69.7
USAAR_DFKI Primary	22.69	66.16
USAAR_DFKI*	22.88	66.05
DFKI-MLT Primary (Transf.large)	24.19†	63.4
Baseline	24.24	62.99
DFKI-MLT Contrastive (Transf.base)	24.5†	62.78†
DFKI-MLT Contrastive (LSTM)	25.3	62.1

Image Source: [3]

WMT18 APE Shared Task

- Difference between the baseline and top ranked system: 0.38 TER points

- Results (NMT):

ID	TER (pe)	BLEU (pe)
FBK Primary (MRT)	16.46	75.53
MS_UEdin Primary	16.5	75.44
FBK Contrastive (MRT+MLE)	16.55	75.38
POSTECH Contrastive (top1)	16.7†	75.14
POSTECH Primary (fix5)	16.71†	75.13
POSTECH Contrastive (var5)	16.71†	75.2
Baseline	16.84	74.73
USAAR_DFKI Primary	17.23	74.22
DFKI-MLT Contrastive (Transf.base)	18.84	70.87
DFKI-MLT Primary (Transf.large)	18.86	70.98
DFKI-MLT Contrastive (LSTM)	19.88	69.35

Image Source: [3]

WMT18 APE Shared Task

- Findings:
  - The difficulty of the APE task: proportional to quality of machine translation system.
  - It is easier to improve translations generated from a PBMT system, using neural APE. But, this is not the case when underlying MT is the NMT system.
  - Systems used the eSCAPE and Artificial corpora. The good improvements in results can be attributed to use of huge amount of synthetic data, along with the technology developments.

WMT19 APE Shared Task

- Language pairs and domain: English-German, English-Russian (IT)
- Underlying MT system: Neural
- Data for English-German: datasets released in earlier rounds
- Data for English-Russian: Train (15,089), Dev (1,000), Test (1,023)
- Number of submissions: 18 (English-German), 4 (English-Russian)
- The quality of English-Russian MT system was much higher: 76.20 BLEU score. None of the systems beat the baseline.
- The top ranked system in the English-german task:
  - Explored transfer learning: Adapted BERT-based encoder-decoder model to APE
  - Used the shared encoder

WMT19 APE Shared Task

- Difference between
  - the baseline and top ranked system: 0.78 TER points

- Results (English-German):

ID	TER (pe)	BLEU (pe)
UNBABEL Primary	16.06*	75.96
POSTECH Primary	16.11*	76.22
POSTECH Contrastive (var2Ens8)	16.13*	76.21
USAAR_DFKI Primary	16.15*	75.75
POSTECH Contrastive (top1Ens4)	16.17*	76.15
UNBABEL Contrastive (2)	16.21*	75.7
UNBABEL Contrastive (1)	16.24*	75.7
FBK Primary	16.37*	75.71
FBK Contrastive	16.61†	75.28
UDS Primary	16.77†	75.03
IC_USFD Contrastive	16.78†	74.88
UDS Contrastive (Gaus)	16.79†	75.03
UDS Contrastive (Uni)	16.80†	75.03
IC_USFD Primary	16.84†	74.8†
Baseline	16.84	74.73
ADAPT_DCU Contrastive (SMT)	17.07	74.3
ADAPT_DCU Primary	17.29	74.29
USAAR_DFKI Contrastive	17.31	73.97
ADAPT_DCU Contrastive (LEN)	17.41	74.01

Image Source: [4]

WMT19 APE Shared Task

- Difference between
  - the baseline and top ranked system: -0.43 TER points

- Results (English-Russian):

ID	TER (pe)	BLEU (pe)
Baseline	16.16	76.2
ADAPT_DCU Contrastive	16.59	75.27
ADAPT_DCU Primary	18.31	72.9
FBK Primary	19.34	72.42
FBK Contrastive	19.48	72.91

Image Source: [4]

WMT19 APE Shared Task

- Findings:
  - The same test as previous year was used for the English-German subtask. Four systems were able to beat the top system of the last year.
  - The quality of translations provided for the English-Russian task was very high. Also, around 60% of the sentences required no edit. This made the problem more challenging.
  - This highlights the problem of over-corrections.

WMT20 APE Shared Task

- Language pairs and domain: English-German, English-Chinese (Wikipedia)
- Data sizes: Train (7,000), Dev (1,000), Test (1,000)
- Number of submissions: 11 (English-German), 4 (English-Chinese)
- Tradeoff: general domain vs lower quality translations
- The top ranked system:
  - Followed the architecture of the last year's winning team
  - Instead of using models like BERT, pretrained a NMT model
  - Used bottleneck adapter layers to prevent overfitting
  - Used external MT candidates. So, the input to APE looked like (SRC, MT_OP, EXT_MT_OP)

WMT20 APE Shared Task

- Difference between
  - the baseline and top ranked system: 11.35 TER points

- Results (English-German):

		TER	BLEU
en-de	HW-TSC_DIRECT_CONTRASTIVE.pe	20.21	66.89
	HW-TSC_CONCAT_PRIMARY.pe	20.52	66.16
	MinD-mem_enc_dec_post-CONTRASTIVE	26.99	55.77
	POSTECH-ETRI_XLM-Top4Ens_CONTRASTIVE	27.02	56.37
	MinD-mem_enc_dec-PRIMARY	27.03	55.58
	POSTECH-ETRI_XLM-Top3Ens_PRIMARY	27.37	55.83
	BeringLab_model1_PRIMARY	27.61	54.71
	BeringLab_model2_CONTRASTIVE	27.96	54.60
	POSTECH_TERNoise-nFold-Ens8_CONTRASTIVE	28.22	54.51
	POSTECH_TERNoise-Ops-Ens8_PRIMARY	28.41	54.22
	Baseline	31.56	50.21
	KAISTxPAPAGO_EMT_PRIMARY	32.00	49.21

Image Source: [5]

WMT20 APE Shared Task

- Difference between
  - the baseline and top ranked system: 12.13 TER points

- Results (English-Chinese):

	TER	BLEU
HW-TSC_CONCAT_PRIMARY.pe	47.36	37.69
HW-TSC_DIRECT_CONTRASTIVE.pe	48.01	37.32
POSTECH-ETRI_XLM-Top3Ens_PRIMARY	54.92	28.90
POSTECH-ETRI_XLM-Top4Ens_CONTRASTIVE	55.08	28.97
Baseline	59.49	23.12

Image Source: [5]

WMT21 APE Shared Task

- Language pairs and domain: English-German, English-Chinese (Wikipedia)
- Data sizes: Train (7,000), Dev (1,000), Test (1,000)
- The data is re-translated and so re-post-edited to improve the quality.
- Number of submissions: 4 (English-German)
- Results:

ID	TER	BLEU
Netmarble_Contrastive	17.28	71.55
PVIE_Contrastive	17.74	70.54
PVIE_Primary	17.85	70.50
Netmarble_Primary	17.97	70.53
Baseline	18.05	71.07

WMT APE Shared Tasks: Summary

Year	2015	2016	2017	2017	2018	2018	2019	2019	2020	2020	2021	2021
Language	En-Es	En-De	En-De	De-En	En-De	En-De	En-De	En-Ru	En-De	En-Zh	En-De	En-Zh
Domain	News	IT	IT	Medical	IT	IT	IT	IT	IT	IT	Wiki	Wiki
MT Type	PBSMT	PBSMT	PBSMT	PBSMT	PBSMT	NMT	NMT	NMT	NMT	NMT	NMT	NMT
Baseline TER	22.91	24.76	24.48	15.55	24.24	16.84	16.84	16.16	31.56	59.49	18.05	-
ΔTER	-0.32	3.24	4.88	0.26	6.24	0.38	0.78	-0.43	11.35	12.13	0.77	-

- ΔTER = Baseline TER - TER of the top-ranked system

- Requirement of post-edits from professional post-editors
- Domain of the data
- Type of underlying MT system and Quality of translations
- Technology Development: Utilization of more and more data
- Uncertainty about effectiveness of current neural approaches

HW-TSC's APE System

- Winner of WMT20 APE Shared Task [8].
- Used transfer learning: Fine-tuned a pre-trained NMT model on the in-domain APE data
- Used data augmentation: translated the source sentences in the APE data using Google's MT system in order to increase diversity of features.
- To control the issue of over-fitting, 'Bottleneck Adapter Layers' are used.
  - It is a low-dimensional FNN layer.

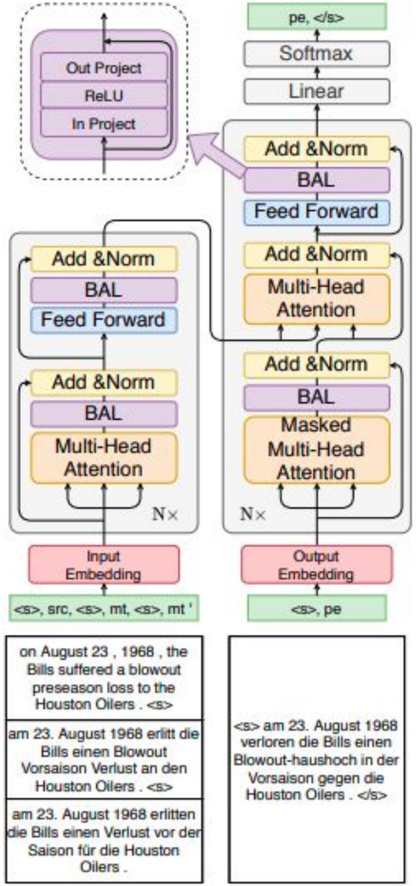


Image Source: [8]

HW-TSC's APE System

- Data used for training NMT model: WMT19 news translation dataset for English-German, and WMT20 news translation dataset
- APE Data: 7000 triplets (wikipedia domain)
- Results on the development set (1000 triplets):

System	En-De BLEU	En-De TER	En-Zh BLEU	En-Zh TER
baseline	50.37	31.374	22.62	60.417
+ Fine-tuning	59.51	25.941	31.74	49.257
+ External MT	65.72	20.959	37.37	47.830
+ Ensemble	66.96	20.222	37.83	46.918
Submission	66.89	20.21	37.69	47.36

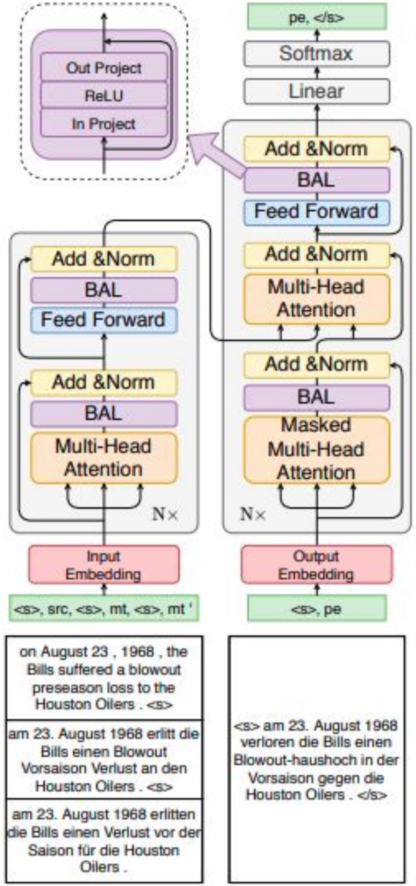


Image Source: [8]

Experiments and Results

Experiment 1

- Goal: To compare different Neural APE approaches.
- We have a English-Marathi parallel corpora but do not have corresponding human post-edits. So we generate an artificial data which can be used to train and evaluate the APE models.
- APE Data Generation : We used two different methods to generate the artificial triplets (SRC, MT_OP, PE_REF) using the Legal domain data from the Anuvaad corpora.
  - Using MT_REF as PE_REF
    - Triplet: (SRC, MT_OP, MT_REF)
  - Using round-trip translation
    - Used Marathi-English MT system to translate MT_REF to SRC’
    - Used English-Marathi MT system to translate SRC’ to MT_OP’
    - New triplet is formed as (SRC’, MT_OP’, MT_REF)
    - Found that the translations generated using this method are noisy.
- Data size: Training: 1.5 lac, Validation: 15k, Testing: 15k

Experiment 1

- APE Systems:
  - Single-source:
    - Treats the APE task as a monolingual translation task.
    - Ignores the SRC. Input to the APE system is MT_OP, and produces PE_REF.
  - Multi-Source:
    - Exploits dependency of errors in translation on the source sentence and the corresponding translation.
    - Input to the system is a (SRC, MT_OP) pair, and outputs the PE_REF.
    - The method uses two separate encoders: one encodes the SRC and the other encodes the MT_OP. And then, a single decoder produces the PE_REF.

Experiment 1

- APE Systems:
  - Multi-source (with shared encoder)
    - This setting uses a single encoder to encode both the SRC and MT_OP.
    - It is beneficial when both languages share the vocabulary.
    - SRC and MT_OP are concatenated and passed to the encoder as a single sequence.
  - Neural Programmer Interpreter
    - Instead of following an end-to-end approach that directly generates a edited sentence from the translation, the method follows a two step approach.
      - In the first phase, the translation is mapped to a sequence of edit operations (insert, keep, delete), and then this sequence along with the translation is used to generate a post-edited sentence.

Experiment 1

- Note:
  - Training data: 1.5L triplets
  - No human post-edited data
  - Underlying NMT system is trained on multi-domain data
  - Used 1000 PE-REF segments to get 'N-modified'

APE System	BLEU	TER	N-modified
Baseline (No APE)	38.54	41.90	-
Single Source	34.86	45.55	50
Multi-source	37.25	42.37	42
Multi-source (Shared Encoder)	34.50	46.23	53
Neural Programmer-Interpreter	27.22	53.57	67

Experiment 1

- Analysis:
  - Not able to find any APE system-specific patterns
  - General observation: Unnecessary edits are performed by the APE systems.
  - Example (Using the Multi-source APE model)
    - SRC: Special powers in case of urgency.
    - MT_REF (and PE_REF): निकडीच्या बाबतीत विशेष अधिकार.
    - MT_OP: तातडीच्या बाबतीत विशेष शक्ती.
    - PE_OP: तातडीच्या परिस्थितीत विशेष शक्ती.

Experiment 2

- Language Pair: English - German
- Data (IT Domain):
  - Combined Training datasets from WMT16, 17, 18 APE shared task
  - Training Data: 36442 Triplets (SRC, MT_OP, PE_REF)
  - Test Data: 1023 Triplets (WMT18 APE Shared Task - NMT)
  - Synthetic Data: 1M Triplets from eSCAPE corpus
- Models:
  - Single Source APE: trained using the synthetic data
  - Multi Source APE: trained using the synthetic data
  - Multi Source APE: trained using the synthetic data + fine-tuned on the Combined Training data

Experiment 2

Model	TER	BLEU
Baseline (No APE)	16.84	74.73
Single-source (Synthetic)	20.02	69.38
Multi-source (Synthetic)	18.89	71.39
Multi-source (Fine-tuned)	16.83	74.73
Neural Program-Interpreter (NPI) (Synthetic)	19.07	71.10
NPI (Fine-tuned)	18.45	72.82

Summary

- From the review of WMT APE Shared, it is clear that the APE can be used in the black-box scenario to further improve quality of translations.
- Recent advancements show a paradigm shift in the field of APE from statistical based APE approaches to neural approaches. This has also given a push to novel ways of synthetic data generation.
- New APE techniques: utilize more and more knowledge.
  - Single-source -- Multi-source -- Pretrained Models -- Data Augmentation
- Whether APE can be used to further improve quality of translations obtained using a high quality NMT system is still unclear.
- The problem of over-correction is prominent when the underlying MT system is of high quality.

References

[1] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes et al. "Findings of the 2016 conference on machine translation." In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, pp. 131-198. 2016.

[2] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck et al. "Findings of the 2017 conference on machine translation (wmt17)." In Proceedings of the Second Conference on Machine Translation, pp. 169-214. 2017.

[3] Rajen Chatterjee, Matteo Negri, Raphael Rubino, Marco Turchi. “Findings of the WMT 2018 Shared Task on Automatic Post-Editing.” In Proceedings of the Third Conference on Machine Translation: Shared Task Papers, pp. 710-725. 2018

[4] Rajen Chatterjee, Christian Federmann, Matteo Negri, and Marco Turchi. "Findings of the WMT 2019 shared task on automatic post-editing." In Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2), pp. 11-28. 2019.

[5] Rajen Chatterjee, Markus Freitag, Matteo Negri, Marco Turchi. Findings of the WMT 2020 Shared Task on Automatic Post-Editing. Proceedings of the Fifth Conference on Machine Translation, EMNLP, Nov. 2020, 646-659, 2020.

References

[6] Junczys Dowmunt, Marcin, and Roman Grundkiewicz. "Log-linear combinations of monolingual and bilingual neural machine translation models for automatic post-editing." In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, ACL. 2016.

[7] Matteo Negri, Marco Turchi, Rajen Chatterjee, Nicola Bertoldi. “ESCAPE: a Large-scale Synthetic Corpus for Automatic Post-Editing.” In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC). 2018.

[8] Yang, Hao, Minghan Wang, Daimeng Wei, Hengchao Shang, Jiaxin Guo, Zongyao Li, Lizhi Lei, Ying Qin, Shimin Tao, Shiliang Sun and Yimeng Chen. “HW-TSC’s Participation at WMT 2020 Automatic Post Editing Shared Task.” WMT 2020.

[9] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. "Sequence to sequence learning with neural networks." Advances in neural information processing systems. 2014.

[10] Bahdanau, Dzmitry, Kyung Hyun Cho, and Yoshua Bengio. "Neural machine translation by jointly learning to align and translate." 3rd International Conference on Learning Representations, ICLR 2015. 2015.

[11] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." Advances in neural information processing systems. 2017.

References

[12] Aakash Banerjee, Aditya Jain, Shivam Mhaskar, Sourabh Dattatray Deoghare, Aman Sehgal, and Pushpak Bhattacharya. 2021. Neural machine translation in low-resource setting: a case study in English-Marathi pair. In Proceedings of Machine Translation Summit XVIII: Research Track, pages 35–47, Virtual. Association for Machine Translation in the Americas.

[13] Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019a. Pivot-based transfer learning for neural machine translation between non-English languages. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 866–876, Hong Kong, China. Association for Computational Linguistics.

[14] Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google’s multilingual neural machine translation system: Enabling zero-shot translation. Transactions of the Association for Computational Linguistics, 5:339–351.

References

[15] Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Speech Synthesis

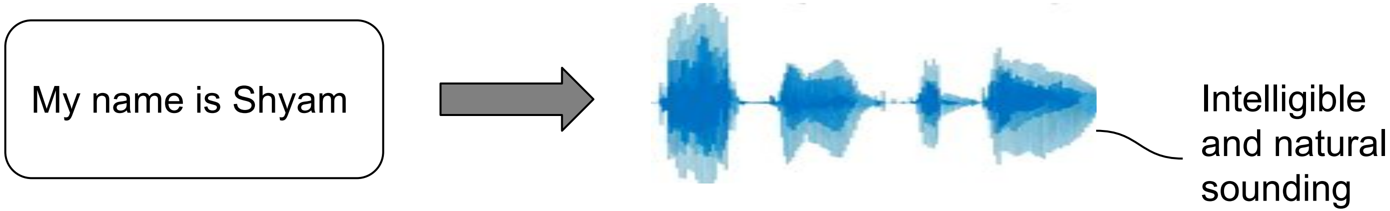
Content

- Introduction
- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Previous Approaches
  - Latest Developments
- Demonstration

Content

- Introduction
- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Introduction
  - Previous Approaches
  - Latest Developments
- Demonstration

Problem Statement



- In this presentation, we focus on **Indian languages**, specifically the Marathi language.
- Also we allow the system to be non-causal, if required.

Motivation (1/2)

Speech synthesis has the potential to improve the daily lives of many people around the world.

- It can give voice to people with speaking disabilities. E.g. People with damaged vocal tract

  The speech synthesiser housing of famous physicist Stephen Hawking

Img source: https://en.wikipedia.org/wiki/Speech_synthesis

Motivation (2/2)

- Machines reading text would enable people with visual impairments or reading disabilities to comprehend any book they wish to read.
  E.g. Audiobooks

- People who are able to understand spoken language, but cannot read text, will get a chance to gain knowledge from the books by *listening* them.
  E.g. Regions in India with high illiteracy rate.

Challenges

- Availability of Data:
  The best English TTS systems are trained on 24 hours of data. The currently available data for Marathi and Hindi is 4.8 hours.

- Text processing:
  Marathi and Hindi are almost phonemic languages. However, currently there do not exist many tools for accurate grapheme-to-phoneme conversion.

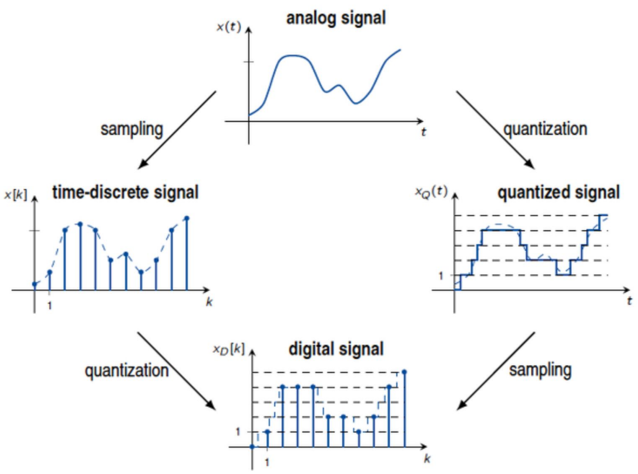
- Handling words from different languages:
  Word sounds from other language do not have much representation in the data, making it difficult to predict.

Content

- Introduction
- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Previous Approaches
  - Latest Developments
- Demonstration

Basics of Signal Processing (1/5)

- Sampling and Quantization
  - Storing analog signal in bits would require infinite memory
  - We discretize the signal along two dimensions - time and value
  - The discretization along the time axis is called as *sampling*. This requires a sampling rate to specified
  - The discretization along the value axis is called as *quantization*. This requires deciding on how many bits we want to use to store one sample of the signal.



Basics of Signal Processing (2/5)

- Digital system
  - Impulse Response:
    - The impulse response of a system is the output that is generated when a delta signal is given as the input
    - It is characteristic of any digital system and provides information about how the system reacts to change in input
  - Output of system:
    - Consider a digital signal **x**[*n*], a digital system with impulse response **h**[*n*], and the output signal **y**[*n*]
    - Then we have,

$$\mathbf{y}[n] = \mathbf{x}[n] * \mathbf{h}[n] = \sum \mathbf{x}[k]\,\mathbf{h}[n-k] \text{ , for all k in } (-\infty, \infty)$$

where, * is the convolution operator

Basics of Signal Processing (3/5)

- Fourier Transform
  - Provides the frequency domain representation of any signal
  - Captures the relative strength of different frequencies in the signal
  - It is an invertible transformation

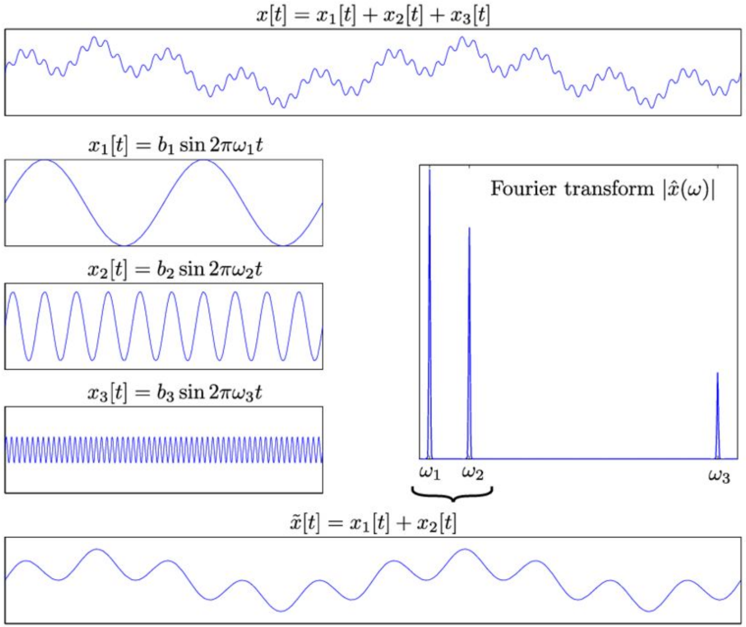
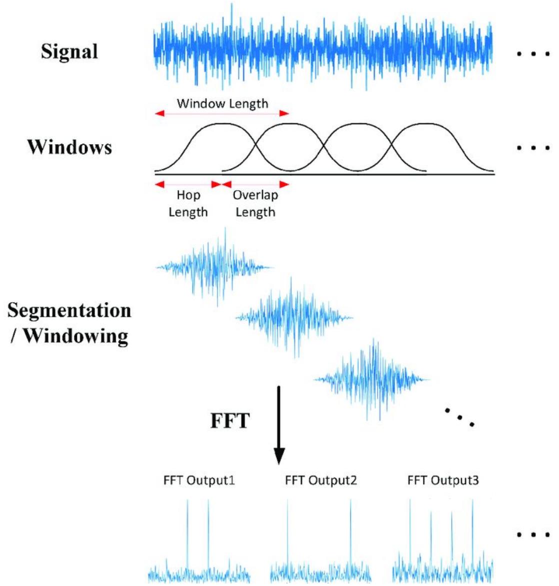


Image source:
https://www.researchgate.net/figure/Fourier-transform-of-a-sum-of-sinusoids-and-filtering-the-highest-frequency_fig3_237061998

Basics of Signal Processing (4/5)

- Discrete Fourier Transform (DFT)
  - Converts a digital signal input of finite length into same-length sequence which represents the samples of fourier transform of the input signal
- Short-time Fourier Transform
  - DFT is applied to small segments of input signal
- Spectrograms
  - The STFT of each segment of the input signal are vertically placed next to each other to form a matrix



Basics of Signal Processing (5/5)

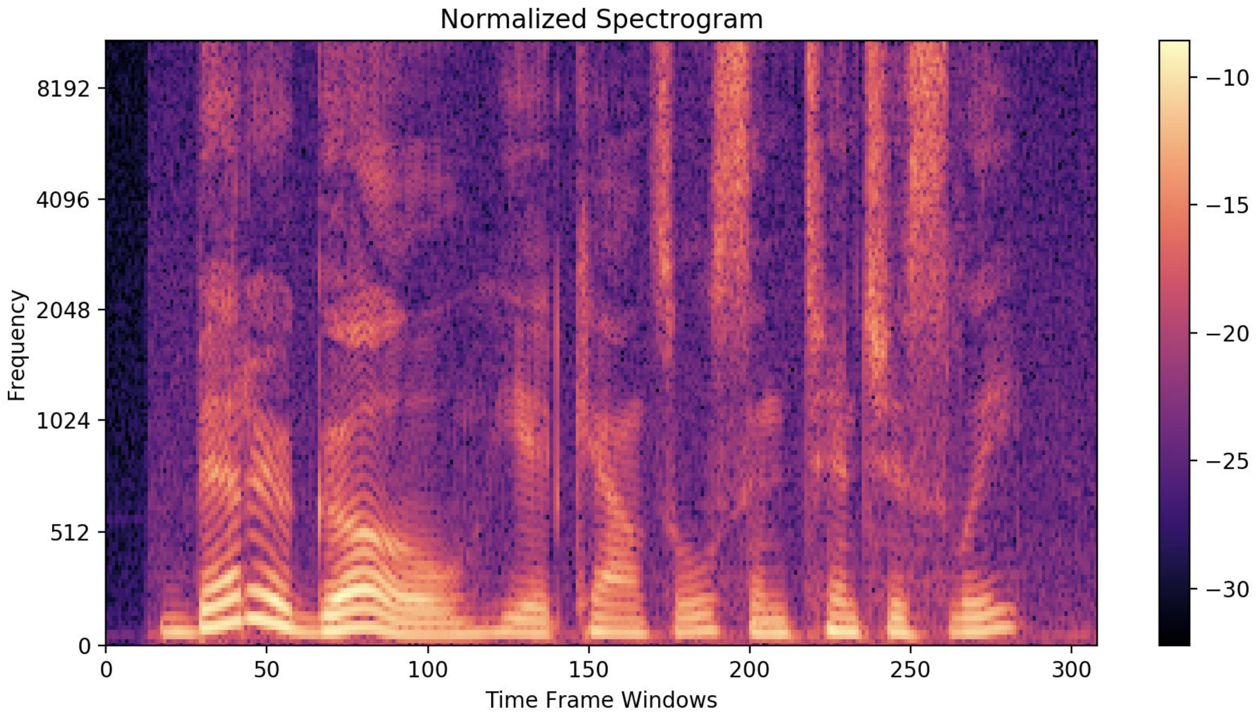


Image source: https://towardsdatascience.com/understanding-audio-data-fourier-transform-fft-spectrogram-and-speech-recognition-a4072d228520

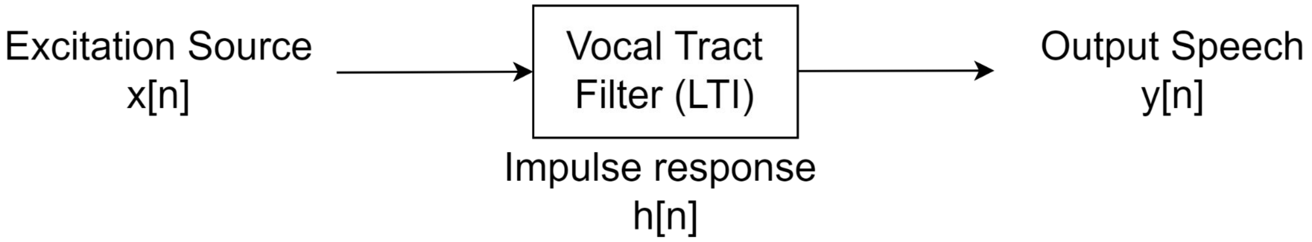
Source-Filter Model (1/3)

- One way to model human speech is via the source-filter approach. Here, the speech is considered to formed by passing a source signal through a filter
  - Source: Periodic vibrations from the glottis (vocal cord)
  - Filter: Vocal tract, including the tongue to change its shape

- Underlying Hypothesis:
  All sounds of any language can be produced by passing a periodic signal through some specific Linear Time-Invariant filter.

Source-Filter Model (2/3)

- Linear Time-Invariant System
  - Linearity: A linear combination of any two signals as input would result in the same linear combination of corresponding outputs.
  - Time-invariant: Shifting the input signal along the time axis, will correspondingly shift the output signal.



Source-Filter Model (3/3)

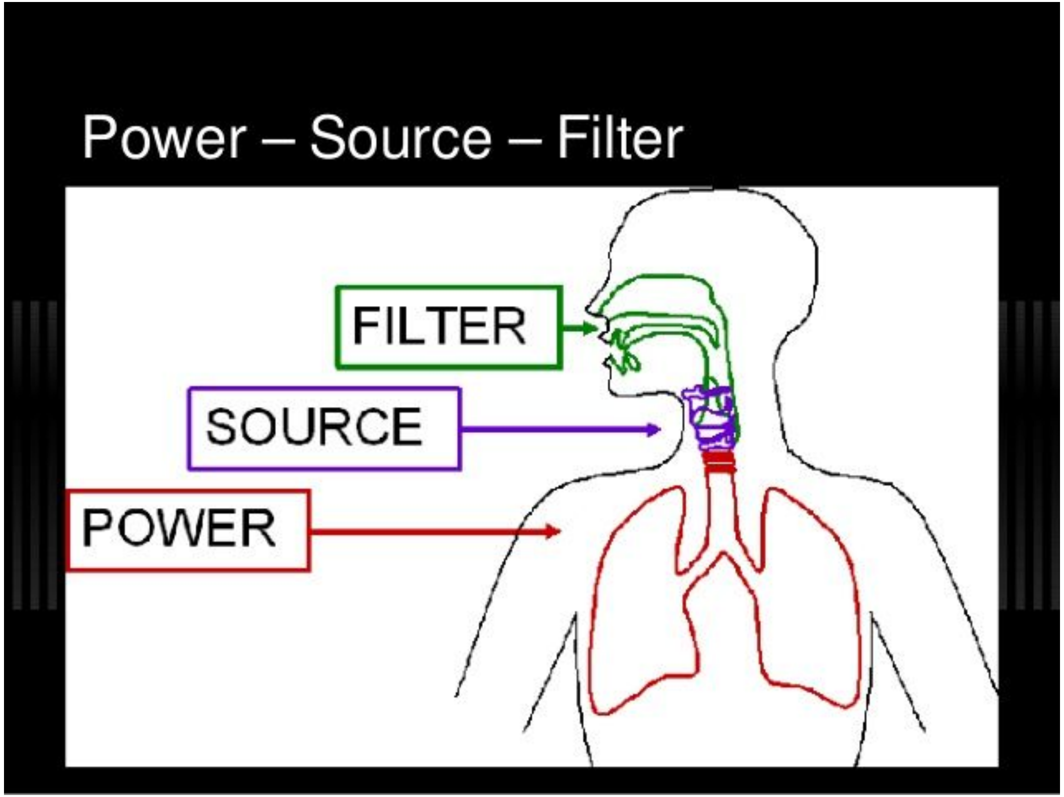


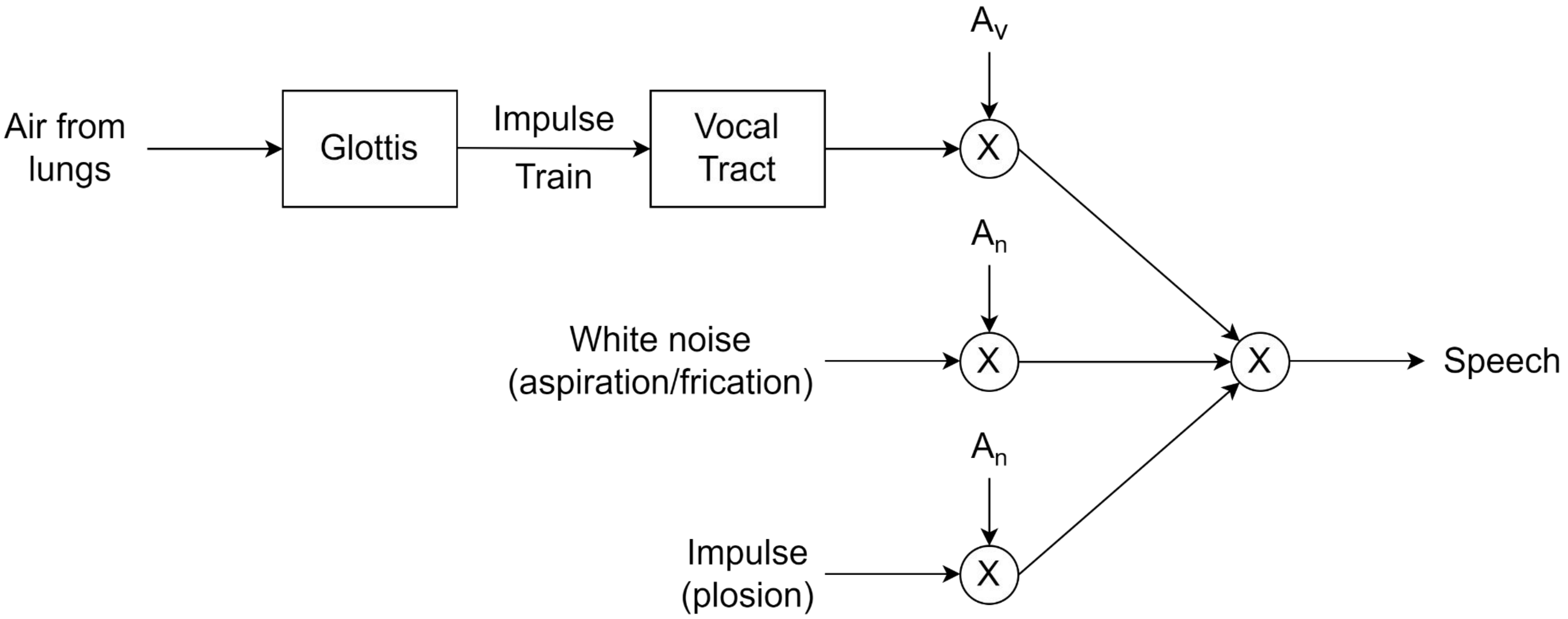
Image source:
https://www.vocalsonstage.com/vocals-on-stage-blog/resonance-and-articulation

Human Speech (1/2)

- Voiced and Unvoiced
  - When the generation of a sound requires vibration of glottis, then we say that it is a *voiced* sound.
    Note: All vowels are voiced sounds.
  - When the vibration of vocal cords are not used for articulation, the generated sound is said to be *unvoiced*.

- Plosives and Fricatives
  - *Plosives* are consonants which require a burst of air source. This is achieved via closure of mouth followed by sudden compression and burst of air (E.g. ब, प)
  - *Fricatives* are consonants which involve a constriction in the vocal tract to create aspiration (E.g. /s/ in sick, /z/ in zebra)

Human Speech (2/2)



Content

- Introduction
- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Previous Approaches
  - Latest Developments
- Demonstration

Perceiving Sound

- Human Hearing (What makes speech sound natural?)
  - Loudness: Proportional to the intensity of the sound
  - Pitch: Human sensation of various frequencies in sound
  - Prosody: Refers to rhythm, stress, and intonation in sound

- Absolute threshold of hearing
  - Minimum sound intensity of a pure tone that can be heard with no other sound in the environment
  - It is frequency dependent (Best range: 2kHz - 5kHz)
  - Measure in dB SPL, which is the pressure relative to 20 micropascal (quietest audible sound pressure level for normal hearing)

Logarithmic Scale and Hearing

- Weber–Fechner law:
  - Above a minimum threshold of perception S_0, the perceived intensity P is logarithmic to the stimulus intensity S

$$P = \mathrm{K} \log(S/S_0)$$

- Empirical evidences support the concept that logarithmic mapping in brain minimizes relative error in perception

- So, a change in intensity at a low intensity is more clearly audible compared to a change in intensity at a high intensity

Source: Varshney, Lav & Sun, John. (2013). Why do we perceive logarithmically?. Significance. 10. 10.1111/j.1740-9713.2013.00636.x.

Mel Frequency Cepstral Coefficient

- Taking the logarithmic hearing into consideration, we can extract MFCC features from any speech signal. The components are:
  - Sampling and Windowing
  - Discrete Fourier Transform
  - Mel Filter Bank
    - Mel frequency scale: $m = 2595 \log_{10}(1 + f / 700)$
  - Discrete Cosine Transform

- Mel Filter Bank:
  Pre-decided number of triangular bandpass filters separated according to the mel frequency scale within specified minimum and maximum frequencies

MFCC Extraction

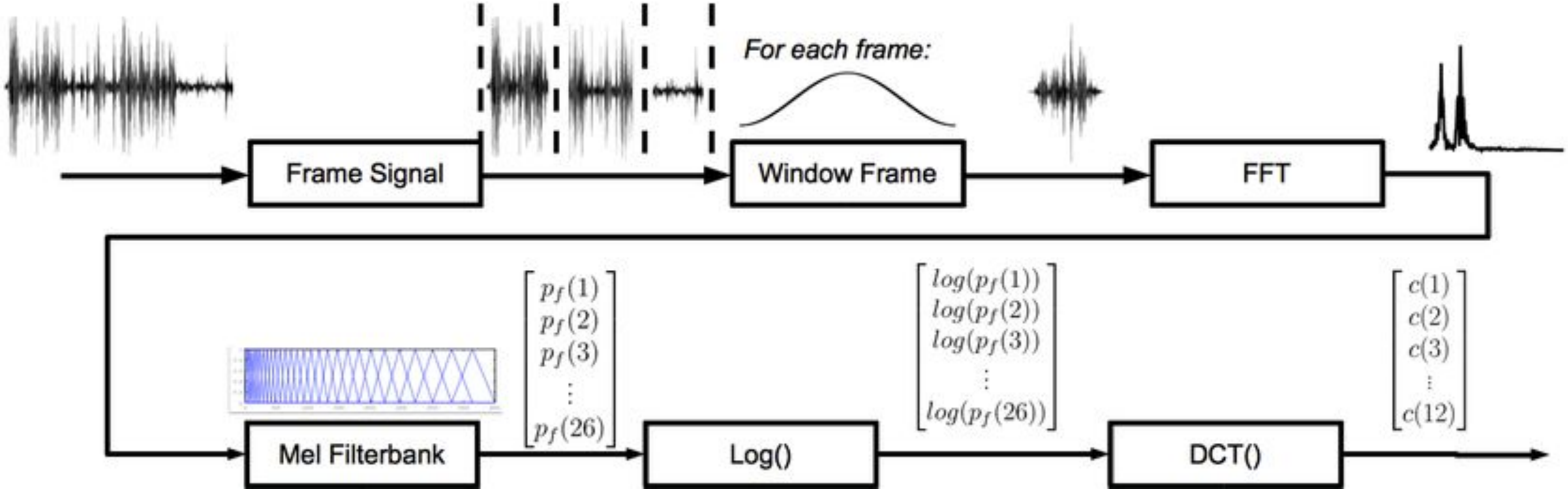


Image source: https://www.researchgate.net/figure/Mel-frequency-cepstrum-coefficient-MFCC-calculation_fig4_277553387

Content

- Introduction
- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Previous Approaches
  - Latest Developments
- Demonstration

History

- 1779 – Mechanical model was built that modelled the human vocal tract, and produced the five vowel sounds in English
- 1791 – Models of tongue and lips were added which allowed production of consonants along with vowels
- 1939 – Keyboard operated voice-synthesizer (Bell Labs)
- Late 1950s – First computer-based speech-synthesis systems

Diphone-based Synthesis (1/3)

- What is a phoneme?
  - The simplest, perceptually distinct units of sound in any language that clearly distinguish one word from another.
- What is a diphone?
  - A combination of two half-phones is called as a diphone.
- The core idea of diphone-based approaches is to decompose speech at the level of diphones and then combine these units based on the characters of input text.

Diphone-based Synthesis (2/3)

- Pitch Synchronous Overlap and Add:
  - Speech is divided into pitch-synchronous waveforms
  - These are then varied in time or spectral domain to obtain synthetic versions of same speech unit
  - Next, the newly formed speech units are overlapped and added to generate the new speech
  - Ideally, there would be no information loss

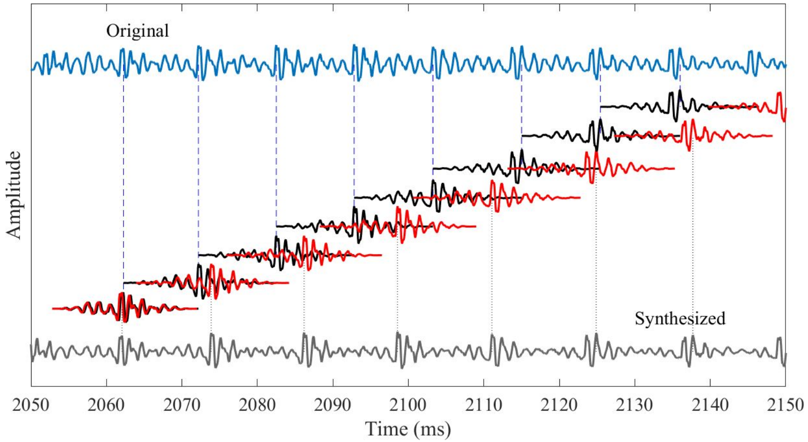


Image Source:
https://wiki.aalto.fi/pages/viewpage.action?pageId=155477136

Diphone-based Synthesis (3/3)

- TD-PSOLA (Time Domain - PSOLA):
  - The speed and pitch of a sound is manipulated in time domain
  - Pitch periods are extracted from the sound signal
    1. Pitch change: Segments are brought closer or separated
    2. Speed change: Segments are repeated or deleted

- FD-PSOLA (Frequency Domain - PSOLA):
  - Spectral envelope is computed using linear predictive analysis
  - Pitch is modified via linear interpolation to obtain synthetic versions of same sound unit
  - Unnatural discontinuities at concatenation boundaries

Unit-Selection Synthesis (1/3)

- Selects & concatenates units (phonemes) from large database
- Text can have additional annotations containing prosodic and phonetic context information
- Database is transformed into a state transition network, with phonemes as states
- The network is fully connected, since any sequence of phonemes is possible

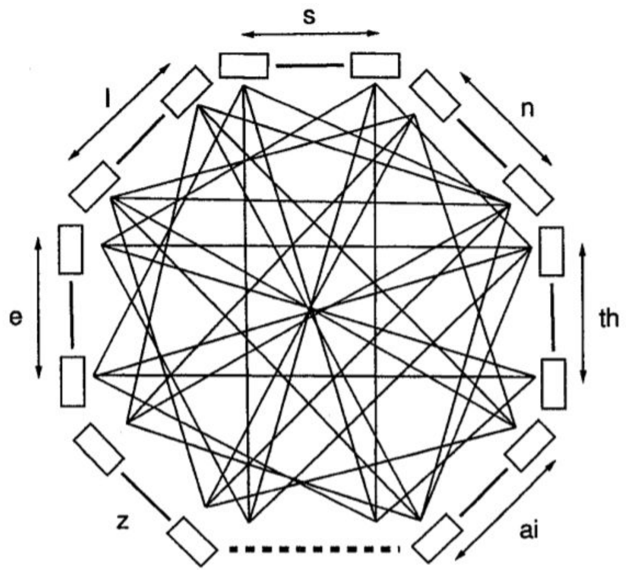


Image source: A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, 1996, pp. 373-376 vol. 1, doi: 10.1109/ICASSP.1996.541110.

Unit-Selection Synthesis (2/3)

- Cost Functions
  - Weighted sum of difference between target and unit feature vectors → weights for each feature need to be learned
  - Target cost:
    - Measures the difference in selected unit and target
    - Feature vectors: pitch, power, duration, voicing, vowel/consonant, consonant type, point of articulation
  - Concatenation cost:
    - Measures the quality of a join between selected units
    - Feature vectors: cepstral distance, difference in log power, pitch

Unit-Selection Synthesis (3/3)

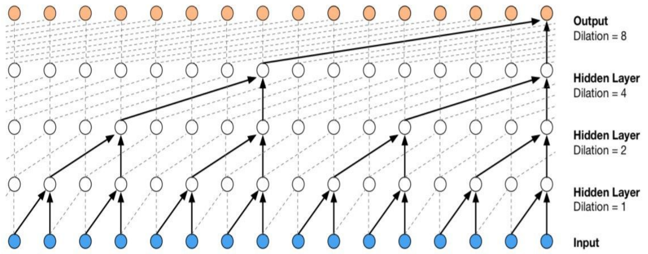
- Learning weights for cost functions
  - Weight Space Search
  - Regression Training

- Generating final speech
  - Using the fully connected graph and the cost functions we can use viterbi decoding for selecting final units
  - The search space for viterbi decoding is pruned based on
    - Phonetic context
    - Target cost
    - Concatenation cost

WaveNet

Wavenet (Google Deepmind)

- Dataset: North American English dataset (24.6 hours)
- First deep neural architecture trained to generate raw audio waveforms that surpassed many approaches of the time.
- Autoregressive generative model
- Model consisted of dilated causal convolutional layers to increase the receptive field of CNN



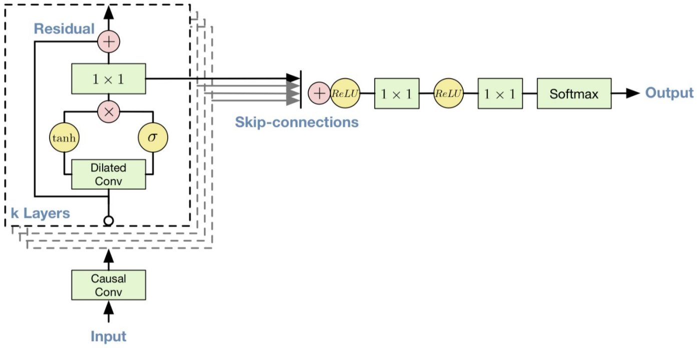
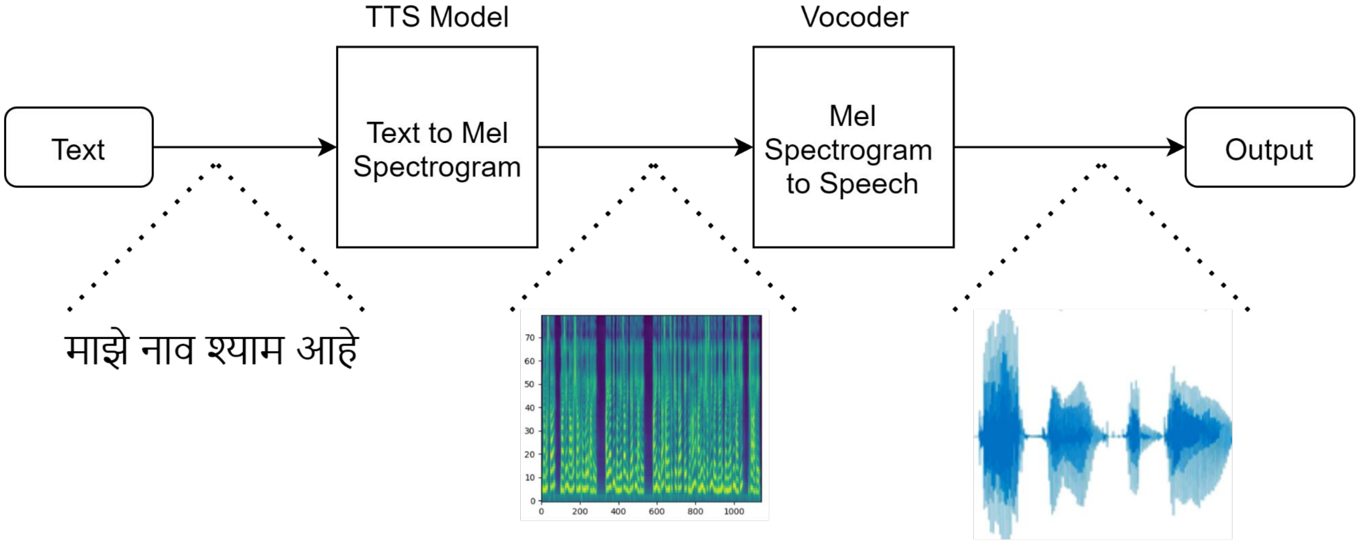


Image source: Oord, Aaron van den, Dieleman, Sander, Zen, Heiga, Simonyan, Karen, Vinyals, Oriol, Graves, Alex, Kalchbrenner, Nal, Senior, Andrew and Kavukcuoglu, Koray WaveNet: A Generative Model for Raw Audio. (2016). , arxiv:1609.03499 .

Content

- Introduction
- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Previous Approaches
  - Latest Developments
- Demonstration

Framework of TTS Systems



All models described ahead follow this pattern for speech synthesis

Text to Mel – Tacotron 2

Tacotron2 (Google)

→ MOS: 3.86

- Dataset: North American English (internal dataset)
- Autoregressive model
- Pre-Net module to bottleneck information flow
- Post-Net module for minor corrections in generated mel spectrogram

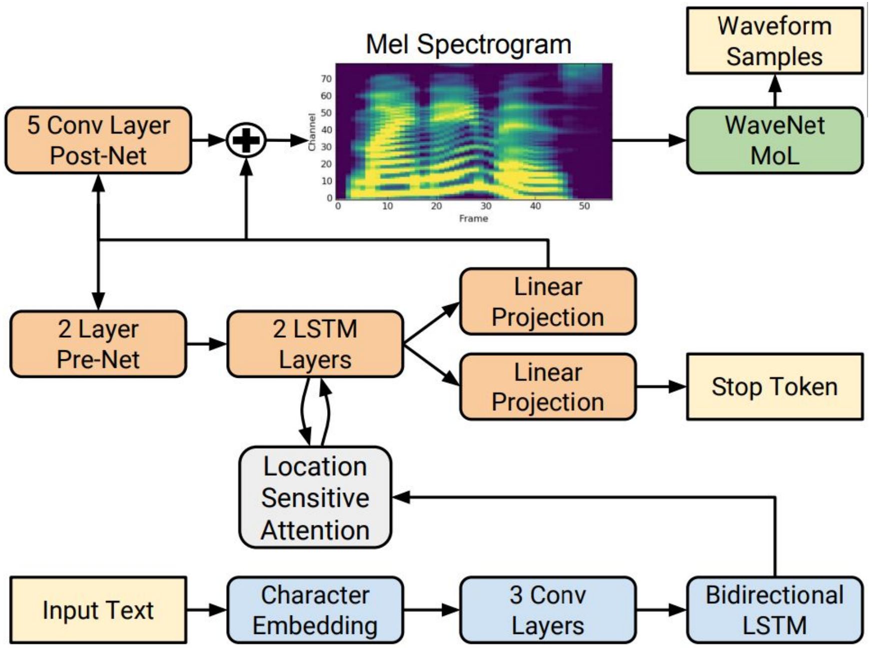
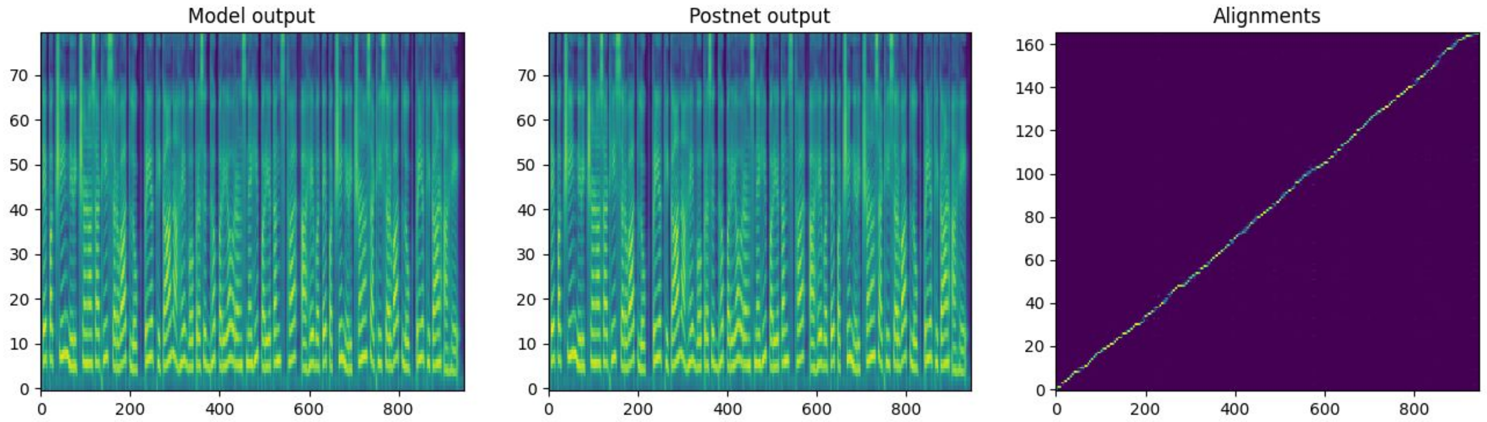


Image source: Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4779–4783. IEEE, 2018.

Perfect attention alignments

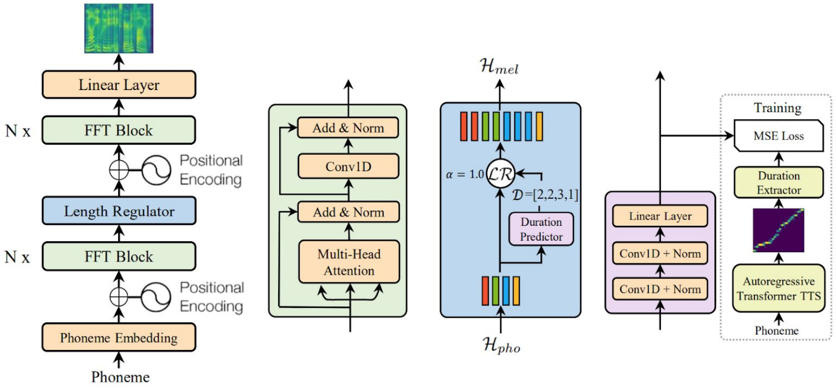


Text to Mel – FastSpeech

FastSpeech (Microsoft)

→ MOS: 3.84

- Dataset: LJSpeech (24 hours)
- Non-autoregressive model
- Feed-Forward Transformer architecture (memory intensive)
- Length regulator to control the length of generated phonemes.



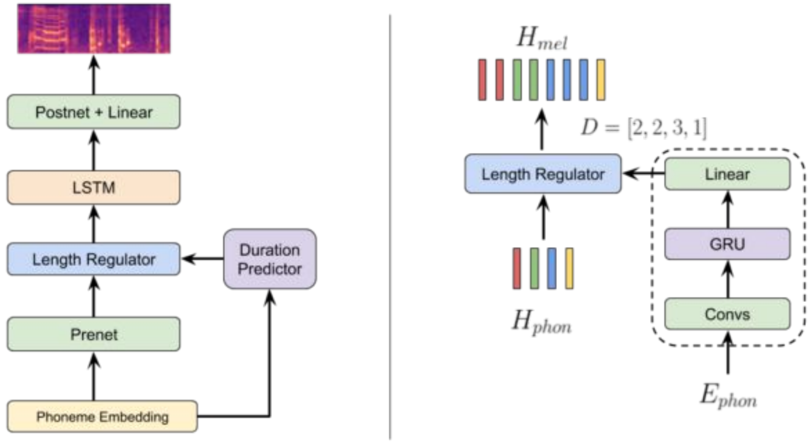
(a) Feed-Forward Transformer (b) FFT Block (c) Length Regulator (d) Duration Predictor

Image source: Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech: Fast, robust and controllable text to speech. In H. Wallach, H. Larochelle, A. Beygelzimer,F. d'Alch´e-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019.

Text to Mel – ForwardTacotron

ForwardTacotron (NVIDIA)

- Dataset: LJSpeech (24 hours)
- Non-autoregressive model
- Tweaked FastSpeech to remove quadratic bottleneck due to self-attention
- Tacotron2 purely used for phoneme duration extraction
- Trained to predict energy and pitch information from ground-truth.



Forward Tacotron

Length Regulator with Duration Predictor

Image source: https://developer.nvidia.com/blog/creating-robust-neural-speech-synthesis-with-forwardtacotron/

Note: There is no paper for ForwardTacotron yet. Above details are taken from a blog released by NVIDIA

Vocoder – WaveGlow

WaveGlow (NVIDIA)
→ MOS: 4.00

- Dataset: LJSpeech (24 hours)
- Flow-based, non-autoregressive generative model
- In flow-based networks, the function approximated by the neural network is invertible
- So if *z* & *x* are the input & output respectively, then

$x = \mathbf{f}_0 \circ \mathbf{f}_1 \circ \ldots \circ \mathbf{f}_k(z)$ & $z = \mathbf{f}_k^{-1} \circ \ldots \circ \mathbf{f}_0^{-1}(x)$

where, $\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_k$ are the layers of the model

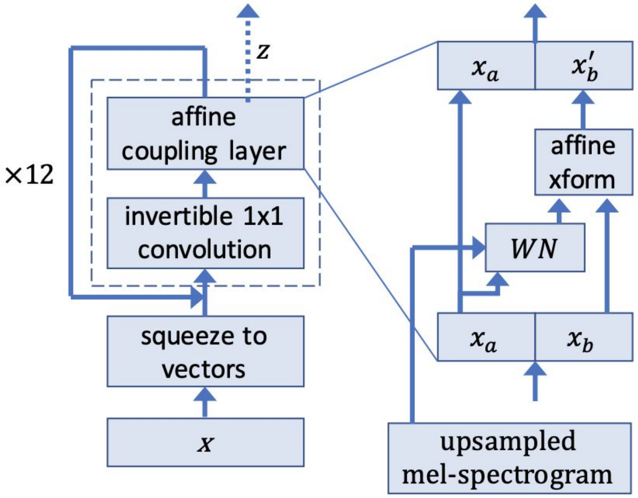
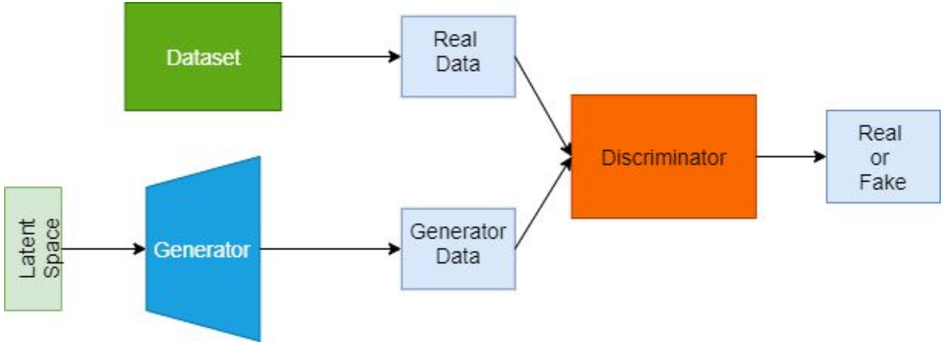


Fig. 1: WaveGlow network

Image source: Ryan Prenger, Rafael Valle, and Bryan Catanzaro. Waveglow: A flow-based generative network for speech synthesis. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3617–3621. IEEE, 2019

Generative Adversarial Networks

- GAN architectures consist of two different models - generator and discriminator
- Generator:
  Generates the desired data from a noise vector
- Discriminator:
  Performs the binary classification task of discriminating between the real data and the data generated by generator



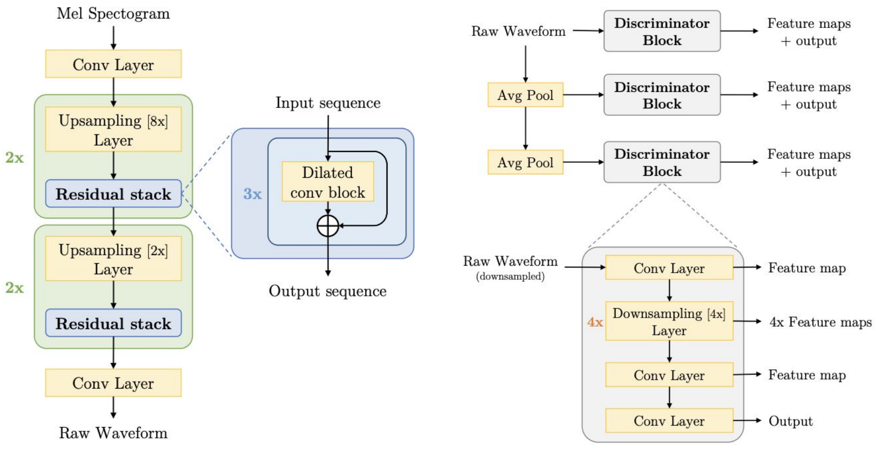
Loss Function:

$$E_x[\log(D(x))] + E_z[\log(1 - D(G(z)))]$$

Vocoder – MelGAN

MelGAN
→ MOS: 3.72

- Dataset: LJSpeech (24 hours)
- First GAN model to produce high quality speech waveforms
- Non-autoregressive and fully-convolutional model
- Multiple discriminators are used to evaluate the input at various resolutions
- Number of parameters 20 times less than Waveglow



(a) Generator (b) Discriminator

Image source: Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, Aaron Courville (2019). MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis.

GAN-based Vocoders

- Recent vocoders have focused on 2 major components
  - GAN architectures
  - Multi-resolution Loss

- Many GAN architectures were introduced in 2020-2021 period
  - ParallelWaveGAN, HiFi-GAN, VocGAN

- All the architectures produced high-quality speech outputs

- A recent paper[11] (Aug 2021) hypothesizes that the performance can be majorly attributed to the multi-resolution discriminative framework, rather than the details of the actual architecture

Content

- Foundations
  - Production of Human Speech
  - Science behind Human Hearing
- TTS Synthesis
  - Introduction
  - Previous Approaches
  - Latest Developments
- Demonstration

Data Description

- Marathi data from IndicTTS Database (developed by IIT Madras) of a female speaker.
- Sampling rate of speech: 22050 Hz

Speech	
Total duration	4.8 hours
Mean duration	7 sec
Standard deviation	2.3 sec
Minimum duration	2 sec
Maximum duration	21 sec

Text	
Total length	24756 words
Mean length	10 words
Standard deviation	2.7 words
Minimum length	4 words
Maximum length	22 words

Data Preparation (1/2)

- Text Cleaners (TC):
  - Simple transliteration cleaner:
    Converts to ASCII using unidecode library
    E.g. माझे नाव श्याम आहे → maajhe naav shyaam aahe
  - Indic transliteration cleaner:
    Uses indic transliteration library (for Indian languages)
    E.g. माझे नाव श्याम आहे → mAjhe nAva shyAma Ahe
- Phonemizer:
  Converts graphemes to phonemes (IPA). Can be used with TC.
    E.g. माझे नाव श्याम आहे → maːɟeː naːv ʃjaːm aːheː

Data Preparation (2/2)

- Mel Spectrogram Generation:

Number of mel filters	80
Minimum mel frequency	0 Hz
Maximum mel frequency	8000 Hz
Filter length	1024
Hop length	256
Window length	1024
Window type	Hanning
Sampling rate	22050 Hz

Experiments and Analysis (1/3)

Less amount of data motivates the use of transfer learning.

- Tacotron2 + WaveGlow (pretrained vocoder):
  - Fine-tuned on Marathi data.
  - Experimented with both text cleaners and phonemizer.

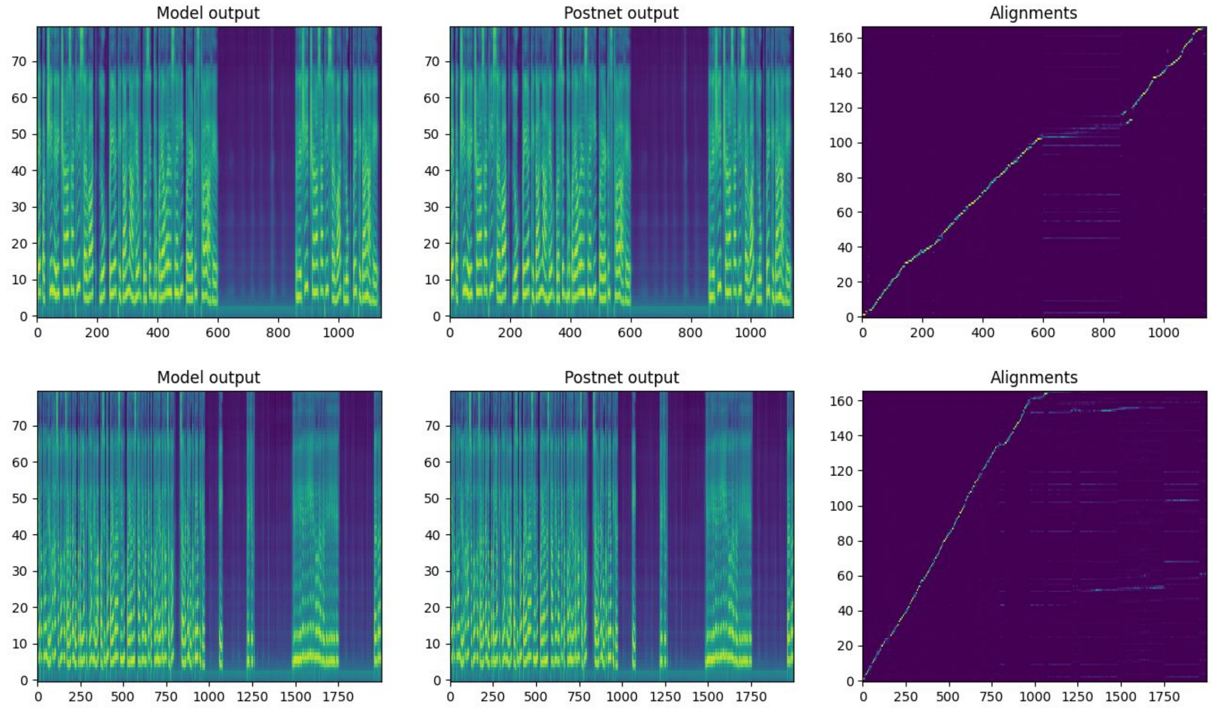
Introduction	नमस्कार, माझे नाव श्याम ठोंबरे आहे, आयआयटी बॉम्बेचा विद्यार्थी आहे, आणि डेटा सायन्स क्षेत्रात एक महान आणि यशस्वी वैज्ञानिक बनण्याची माझी इच्छा आहे.
Hindi and Marathi	नमस्कार, माझे नाव श्याम ठोंबरे आहे, आज जब मैं सुपरमार्केट गया तो देखा कि सब्जी बेचने वाले लोगों ने मास्क नहीं पहना हुआ था.
Pronunciation Problem	मी माझ्या व्यवस्थापकाने मला दिलेली रक्कम जमा करण्यासाठी बँकेत गेलो होतो, परंतु त्यांनी सांगितले की गेल्या 5 दिवसात मिळालेले पैसे त्वरित जमा करण्याची परवानगी नाही.

Experiments and Analysis (2/3)

- Tacotron2 fails on some text inputs – 2 major issues

मी माझ्या व्यवस्थापकाने
मला दिलेली रक्कम जमा
करण्यासाठी बँकेत गेलो होतो,
परंतु त्यांनी सांगितले की
गेल्या 5 दिवसात मिळालेले
पैसे त्वरित जमा करण्याची
परवानगी नाही.

Hazy Alignment and Garbage Output



Experiments and Analysis (3/3)

- ForwardTacotron + WaveGlow (pretrained vocoder):
  - Trained from scratch on Marathi data
  - Experimented with phonemizer

Introduction →	नमस्कार, माझे नाव श्याम ठोंबरे आहे, आयआयटी बॉम्बेचा विद्यार्थी आहे, आणि डेटा सायन्स क्षेत्रात एक महान आणि यशस्वी वैज्ञानिक बनण्याची माझी इच्छा आहे.
Hindi and Marathi →	नमस्कार, माझे नाव श्याम ठोंबरे आहे, आज जब मैं सुपरमार्केट गया तो देखा कि सब्जी बेचने वाले लोगों ने मास्क नहीं पहना हुआ था.
Pronunciation Problem →	मी माझ्या व्यवस्थापकाने मला दिलेली रक्कम जमा करण्यासाठी बँकेत गेलो होतो, परंतु त्यांनी सांगितले की गेल्या 5 दिवसात मिळालेले पैसे त्वरित जमा करण्याची परवानगी नाही.
Challenging Pronunciation →	जयोस्तु ते जयोस्तु ते! जयोस्तु ते! श्री महन्मंगले शिवास्पदे शुभदे स्वतंत्रते भगवती त्वामहम् यशोयुतां वंदे! राष्ट्राचें चैतन्य मूर्त तूं नीती संपदांची स्वतन्त्रते भगवती श्रीमती राज्ञी तूं त्यांची परवशतेच्या नभांत तूंचि आकाशीं होशी स्वतन्त्रते भगवती चांदणी चमचम-लखलखशी.

Hindi Outputs

Introduction	नमस्कार, मेरा नाम श्याम है और मैं आईआईटी बॉम्बे का छात्र हूं जो डेटा साइंस के क्षेत्र में अध्ययन कर रहा है।
Bank statement	जब मुझे इस महीने का वेतन मिला तो मैं बैंक में पैसा जमा करने गया, लेकिन उन्होंने कहा कि पिछले 5 दिनों में मुझे जो पैसा मिला है, उसे मैं जमा नहीं कर सकता।
Aspirated consonants	आज मेस में दोपहर का भोजन वास्तव में स्वादिष्ट था, इसलिए अब मुझे अपने घर के पास खाने के लिए नई जगह ढूंढने की आवश्यकता नहीं है।
Summer in Himalaya	गर्मियों में, कोरोना महामारी समाप्त होने के बाद, मैं उत्तर में हिमालय, दुनिया के सबसे महान पर्वत पर जाना चाहूंगा।

Summary (1/2)

- Text-to-Speech synthesis attempts to generate intelligible and natural sounding speech for any given text
- Many intricacies are involved during human speech production and source-filter model provides great insights into it
- Human hearing which is more sensitive to logarithmic frequency scale
- MFCC features align with the nature of human hearing and are great at capturing the characteristics of human speech

Summary (2/2)

- Earlier approaches created database and smartly concatenated individual phoneme-based units to generate the output speech

- Deep Neural architectures first generate the mel spectrograms from the text which is then converted into waveforms by vocoders

- Recently, GAN approaches have dominated as vocoders, and the adversarial training with multi-resolution training have ensured very high-quality output speech

References

[1] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, 1996, pp. 373-376 vol. 1, doi: 10.1109/ICASSP.1996.541110.

[2] Oord, Aaron van den, Dieleman, Sander, Zen, Heiga, Simonyan, Karen, Vinyals, Oriol, Graves, Alex, Kalchbrenner, Nal, Senior, Andrew and Kavukcuoglu, Koray WaveNet: A Generative Model for Raw Audio. (2016). , arxiv:1609.03499

[3] Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, et al. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4779–4783. IEEE, 2018.

[4] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech: Fast, robust and controllable text to speech. In H. Wallach, H. Larochelle, A. Beygelzimer,F. d'Alch´e-Buc, E. Fox, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc., 2019

References

[5] https://developer.nvidia.com/blog/creating-robust-neural-speech-synthesis-with-forwardtacotron/

[6] Ryan Prenger, Rafael Valle, and Bryan Catanzaro. Waveglow: A flow-based generative network for speech synthesis. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 3617–3621. IEEE, 2019

[7] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, Aaron Courville (2019). MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis.

[8] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, Advances in Neural Information Processing Systems, volume 33, pages 17022–17033. Curran Associates, Inc., 2020.

[9] Ryuichi Yamamoto, Eunwoo Song, Jae-Min Kim, Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6199–6203, 2020

References

[10] Jinhyeok Yang, Junmo Lee, Youngik Kim, Hoonyoung Cho, Injung Kim, Vocgan: a high fidelity real-time vocoder with a hierarchically-nested adversarial network, INTERSPEECH, pp. 200–204, 2020.

[11] Jaeseong You, Dalhyun Kim, Gyuhyeon Nam, Geumbyeol Hwang, and Gyeongsu Chae. GAN Vocoder: Multi-Resolution Discriminator Is All You Need. InProc. Interspeech 2021, pages 2177–2181, 2021

SSMT Demonstration

Conclusions

Conclusions (1/2)

- SSMT is the task to translate speech in language A into speech in language B through use of a computer.
- Applications- movie dubbing, movie Subtitling, conversing with foreign-language speakers, etc.
- Accent, hesitation, disfluency, and grammatically-flexible nature of spoken languages make SSMT more challenging than text-to-text MT.
- The task can either be trained -
  - as Direct Speech to Speech conversion, or
  - Cascaded SSMT. It can be divided into 3 subtasks:
    - Automatic Speech Recognition (ASR)
    - Machine Translation (MT)
    - Text-to-Speech synthesis (TTS)

Conclusions (2/2)

- We illustrated Direct SSMT approaches which work on signal level avoiding intermediate text generation.
- We discussed different ASR and TTS techniques and showed demonstrations.
- We also discussed Disfluency correction, Automatic post-editing techniques which are used to remove irregularities from speech transcriptions to make that ready for MT and vice versa.
- We discussed different paradigms of MT along with recent advancements (*i.e.* LaBSE filtering, Phrase table injection, Pivoting, Multilingual MT, Unsupervised MT). We also demonstrated the MT system.
- We demonstrated the entire SSMT pipeline.

Future directions

- Improving ASR results in Indian languages in a conversational setting and speaker diarization
- Checking usability of current neural APE approaches on improving translations obtained from a high-quality NMT system
- Restricting APE systems from performing unnecessary edits
- Improving disfluency correction with the help of other language data.
- Improving the naturalness of output speech through text processing for Indian languages TTS synthesis

Resources

All the resources will be made available here:

https://github.com/sourabhd13/ssmt_tutorial_icon2021

Thank You!