CS772: Deep Learning for Natural Language Processing (DL-NLP)

Prompting, Reasoning, Bias, SSMT, QE, APE, Fake-News & Half-Truth Detection, Query Intent Detection and Speech Emotion Recognition

Instructor: Prof. Pushpak Bhattacharyya
Computer Science and Engineering Department
IIT Bombay

Week 10 of 13th March, 2023

CS772: Deep Learning for Natural Language Processing (DL-NLP)

Prompting, Reasoning, Ethics in NLP

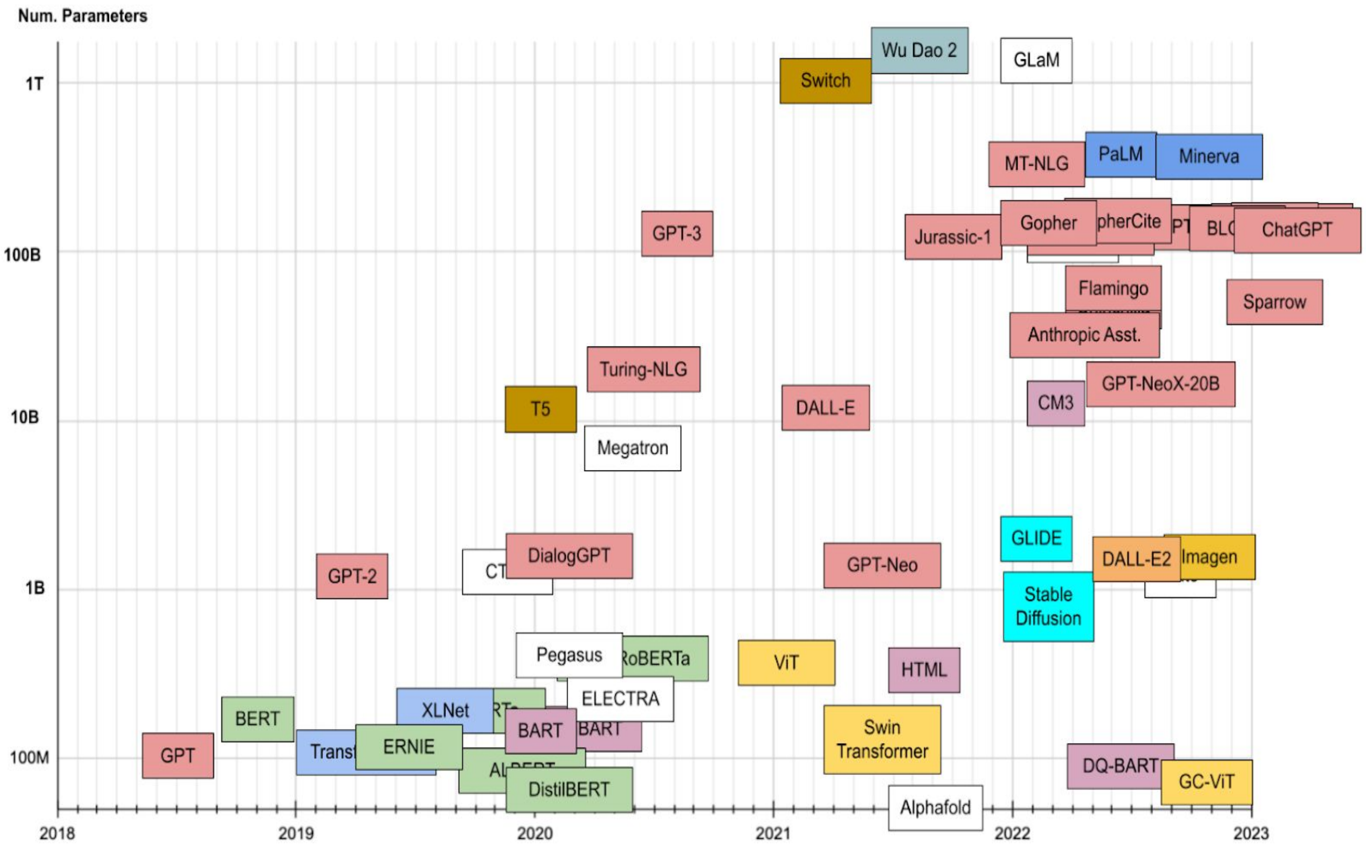
Sravanthi, Nihar
Computer Science and Engineering Department
IIT Bombay

Week 10 of 13mar23

Outline

- Need for Prompting
- Paradigms in NLP
- Terminologies and Notations
- Design considerations for Prompting
    - Pre-trained Model Choice
    - Prompt -Engineering
    - Answer Engineering
    - Expanding the paradigm
    - Prompt-based Training Strategies
- Reasoning with Large Language Prompting
- Ethics in NLP

Need for Prompting



(Increase in LLM parameter space)

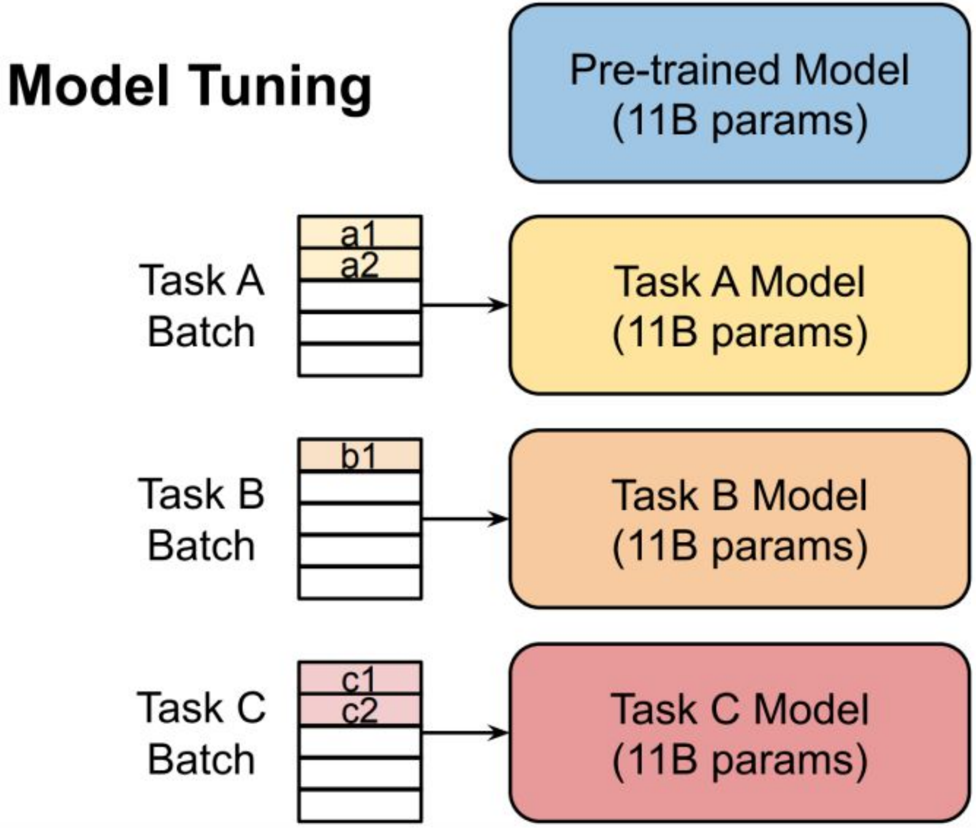
Need for Prompting

- With the increase in size of LLMs fine-tuning becomes infeasible and ineffective.
- Model should be able to predict without any gradient updates.
- We aim to have a single model perform many downstream tasks.
- Given an instruction or, few examples model should understand the task and predict correct answers.

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

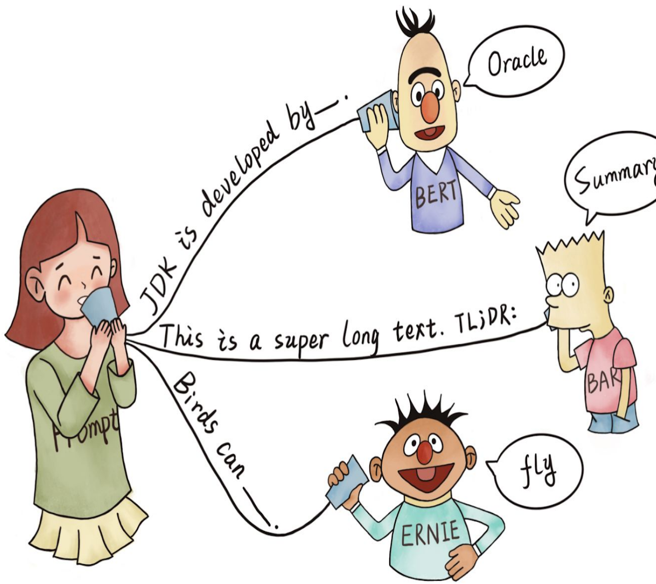
- ELMo: 1B training tokens
- BERT: 3.3B training tokens
- RoBERTa: ~30B training tokens
- PaLM: 750B tokens

Need for Prompting

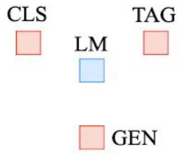
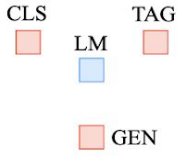
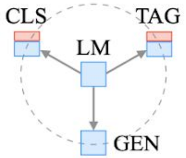
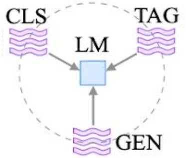


Prompting in Layman's term

- Encouraging a pre-trained model to make particular predictions by providing a "prompt" specifying the task to be done.
- No fine-tuning!!! Literally just take a pretrained LM and give it the following prefix:
- Translate English to French: sea otter => loutre de mer, cheese =>"



Four paradigms in NLP

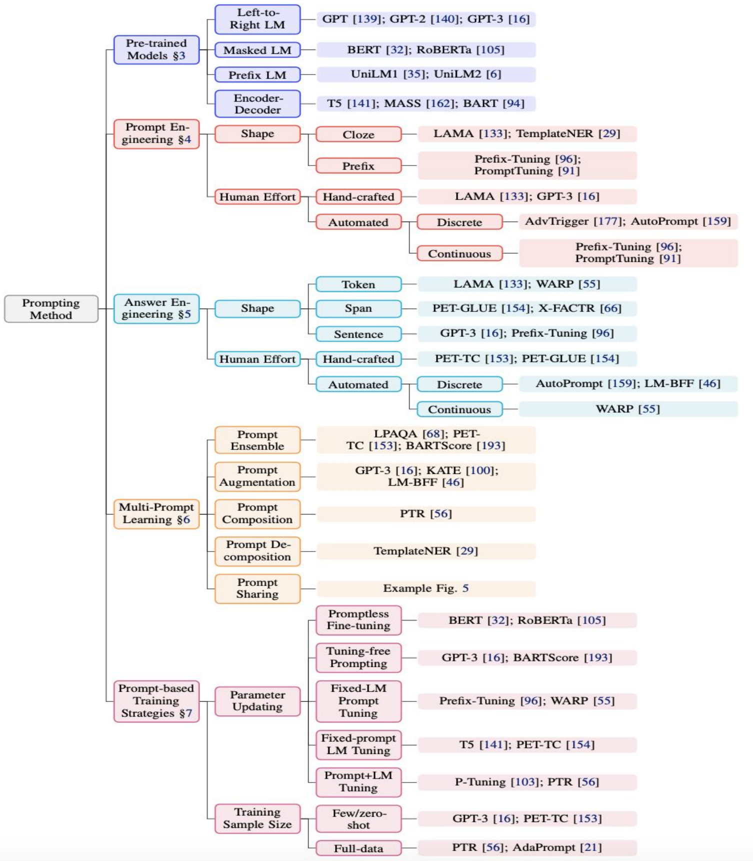
Paradigm	Engineering	Task Relation
a. Fully Supervised Learning (Non-Neural Network)	Features (e.g. word identity, part-of-speech, sentence length)	
b. Fully Supervised Learning (Neural Network)	Architecture (e.g. convolutional, recurrent, self-attentional)	
c. Pre-train, Fine-tune	Objective (e.g. masked language modeling, next sentence prediction)	
d. Pre-train, Prompt, Predict	Prompt (e.g. cloze, prefix)	

Terminology and notation of prompting methods

Name	**Notation**	**Example**	**Description**
Input	$\boldsymbol{x}$	I love this movie.	One or multiple texts
Output	$\boldsymbol{y}$	++ (very positive)	Output label or text
Prompting Function	$f_{\text{prompt}}(\boldsymbol{x})$	`[X]` Overall, it was a `[Z]` movie.	A function that converts the input into a specific form by inserting the input $\boldsymbol{x}$ and adding a slot `[Z]` where answer $\boldsymbol{z}$ may be filled later.
Prompt	$\boldsymbol{x}'$	I love this movie. Overall, it was a `[Z]` movie.	A text where `[X]` is instantiated by input $\boldsymbol{x}$ but answer slot `[Z]` is not.
Filled Prompt	$f_{\text{fill}}(\boldsymbol{x}', \boldsymbol{z})$	I love this movie. Overall, it was a bad movie.	A prompt where slot `[Z]` is filled with any answer.
Answered Prompt	$f_{\text{fill}}(\boldsymbol{x}', \boldsymbol{z}^*)$	I love this movie. Overall, it was a good movie.	A prompt where slot `[Z]` is filled with a true answer.
Answer	$\boldsymbol{z}$	"good", "fantastic", "boring"	A token, phrase, or sentence that fills `[Z]`

Design considerations for Prompting

- Pre-trained Model Choice
- Prompt -Engineering
- Answer Engineering
- Expanding the paradigm
- Prompt-based Training Strategies



Pretrained Language Model Choice

Left-to-right Language Model :

- The earliest architecture chosen for prompting.
- Usually used with prefix prompts and parameters on the LLM are fixed.
- GPT-2, GPT-3, BLOOM

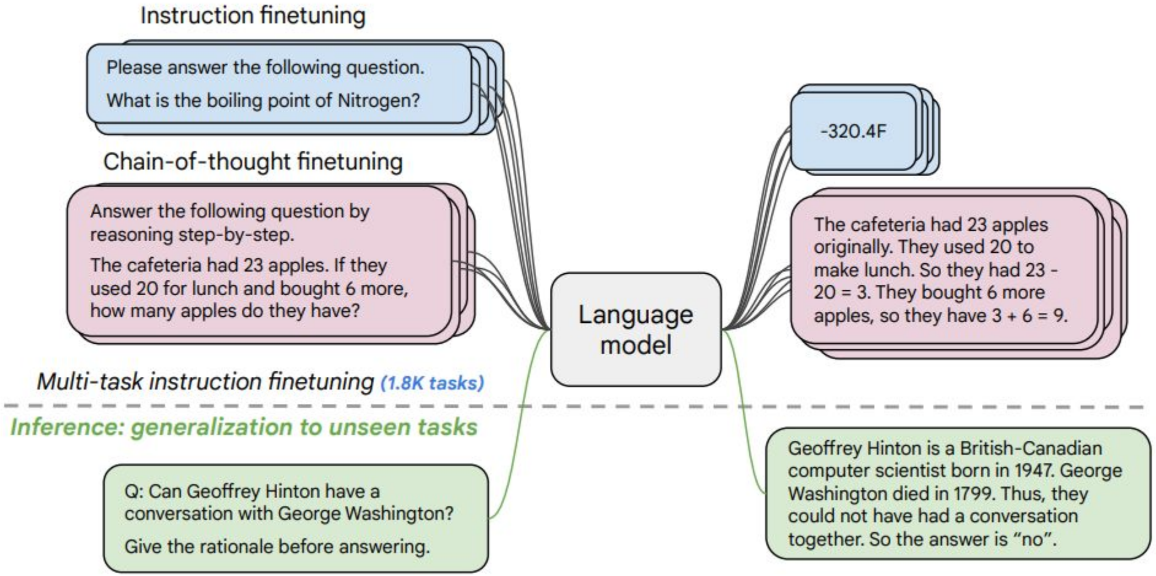
Masked Language Model:

- Usually combined with cloze prompt
- Suitable for NLU tasks, which should be reformulated to cloze tasks.
- BERT, ERNIE

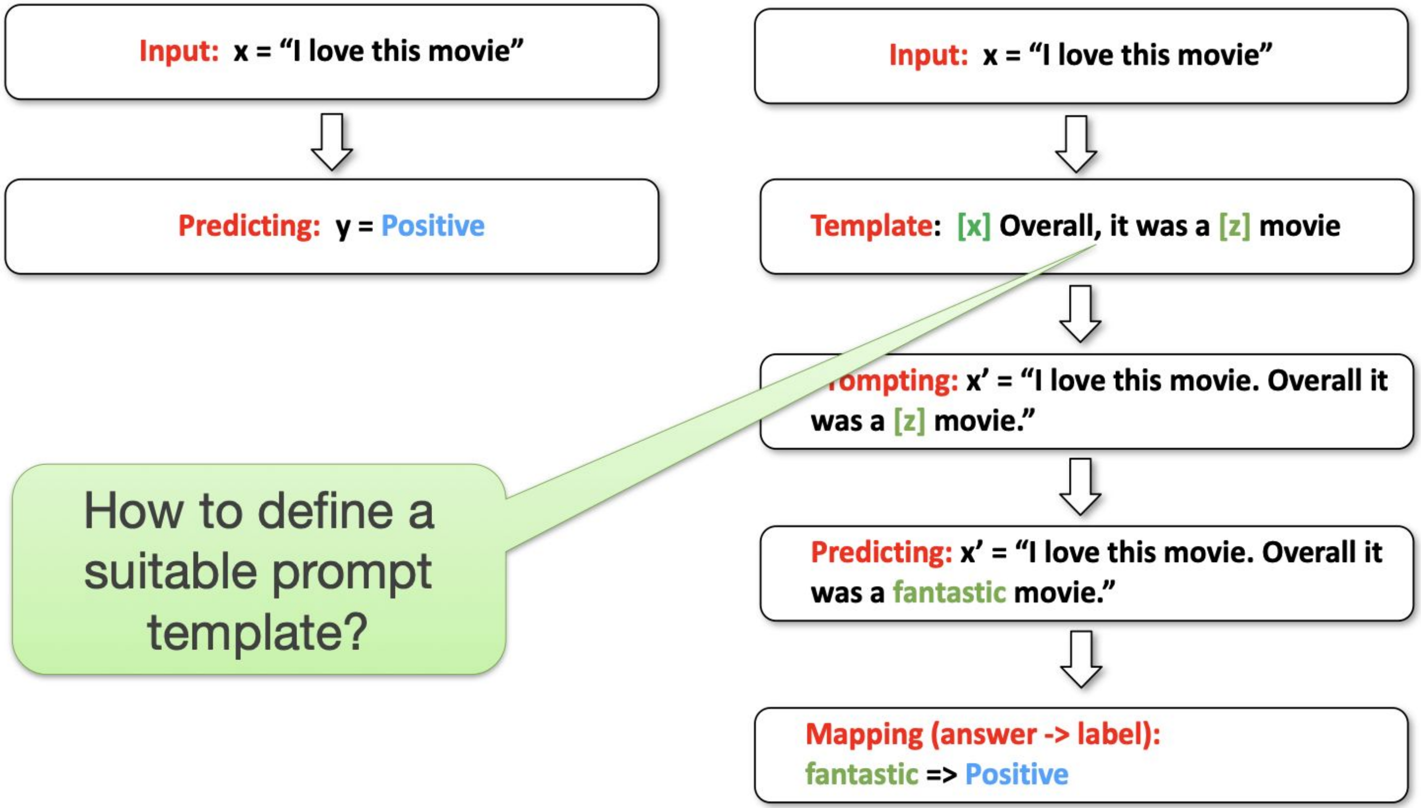
Pretrained Language Model Choice

Instruction-tuned LM:

- The SOTA LLMs created to generalise well on various tasks.
- The LMs are created by 1) scaling number of tasks, 2) scaling the model size, 3) fine-tuning using chain-of-thought data, 4) training using RLHF
- flanT5, InstructGPT, chatGPT



Prompt Engineering - Traditional vs Prompt formulations



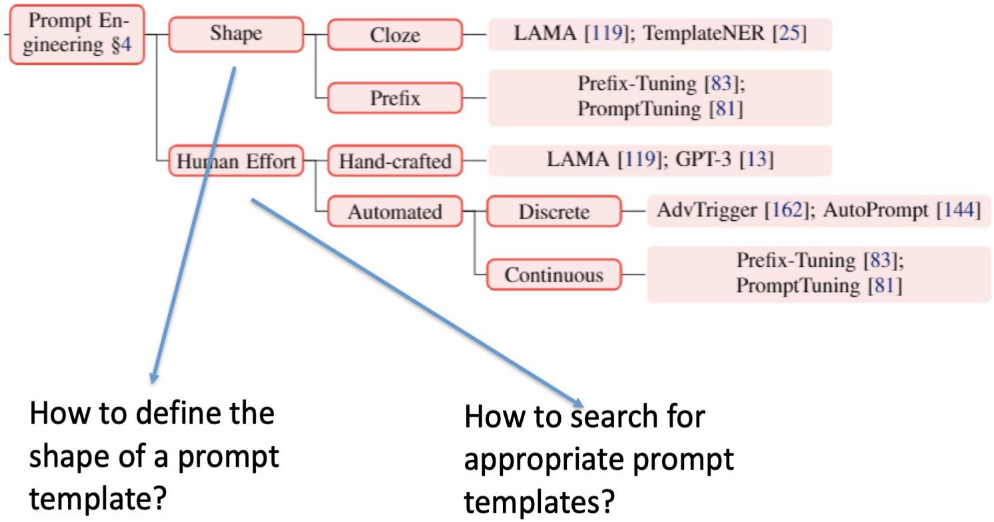
Prompt Template Engineering

Prompt shape:

- Cloze prompt
- Prefix prompt

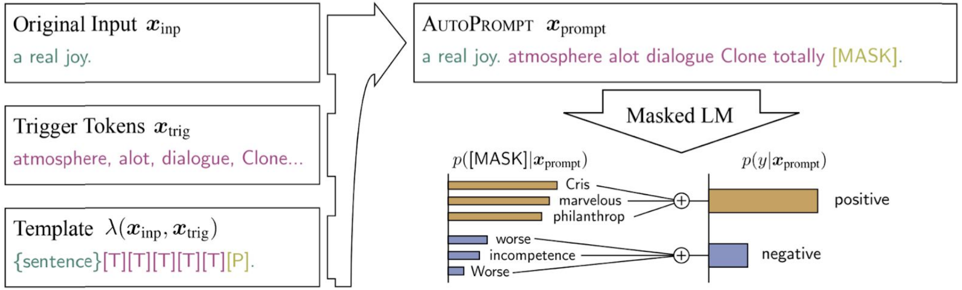
Design of Prompt Template:

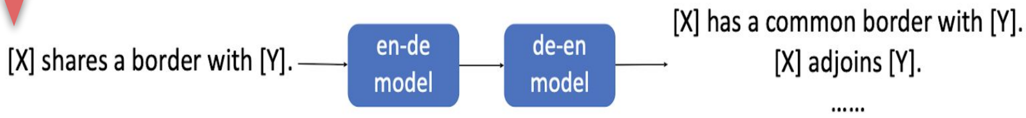
- Hand crafted
- Automated search
  - Search in discrete space
  - Search in Continuous space

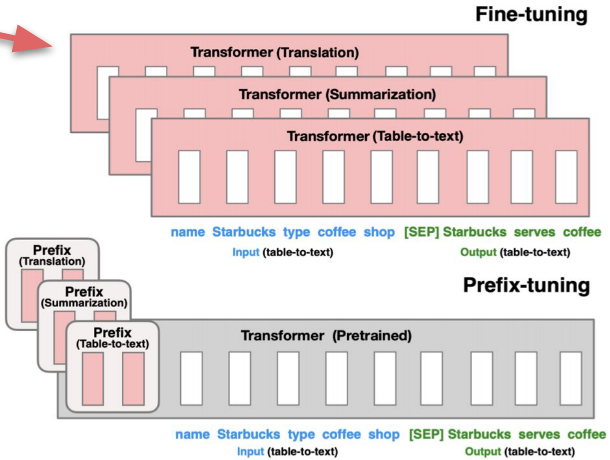


Representative methods for prompt search

- Prompt mining
- Prompt paraphrasing
- Gradient based search
- Prompt/Prefix tuning





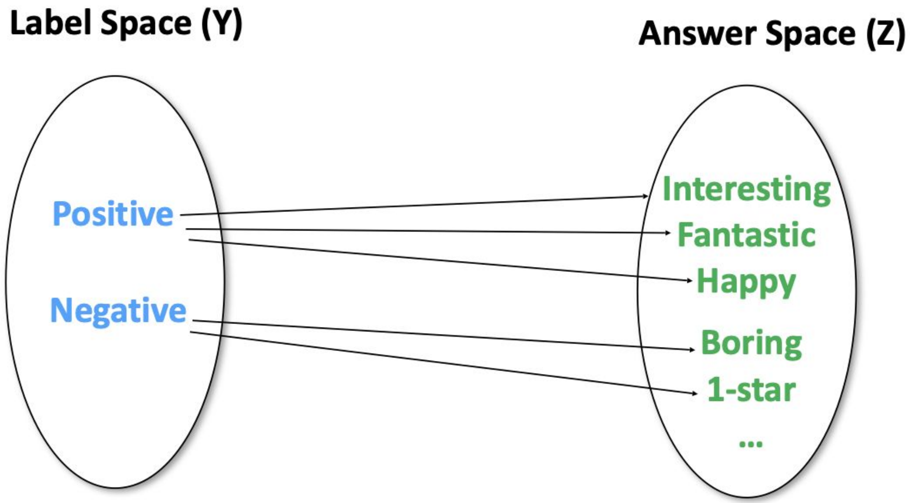


Answer Engineering

Why do we need answer engineering?

- We have reformulate the task! We also should re-define the "ground truth labels"

Definition: aims to search for an answer space and a map to the original output Y that results in an effective predictive model



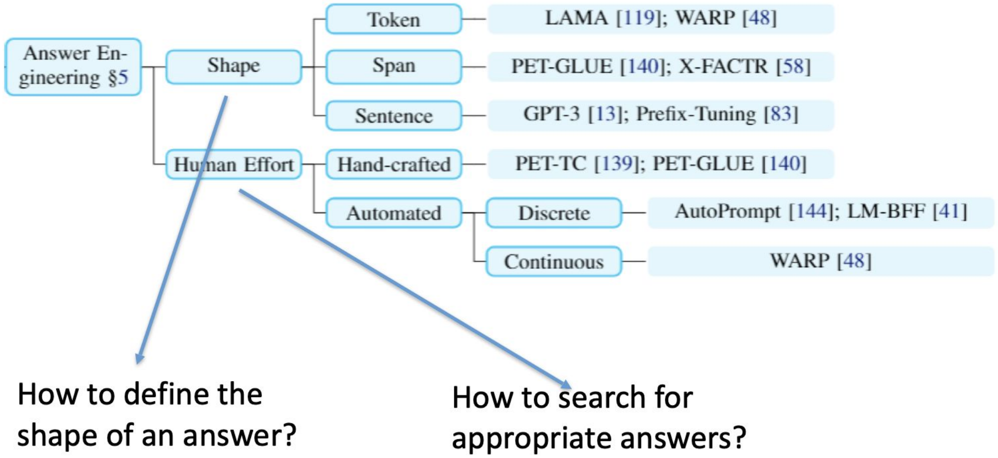
Design of Prompt Answer

Answer Shape:

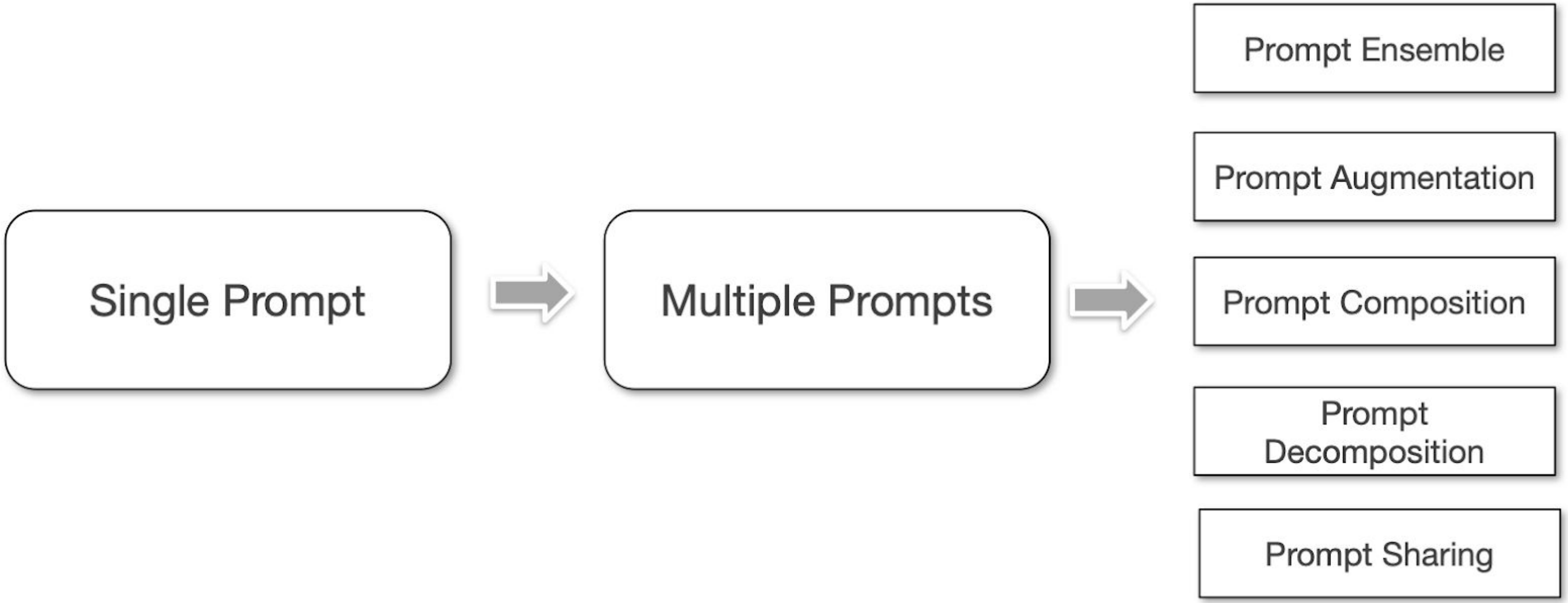
Token, Chunk, Sentence

Answer Search:

- Hand crafted
  - Infinite answer space
  - Finite answer space
- Automated search
  - Discrete space
  - Continuous space



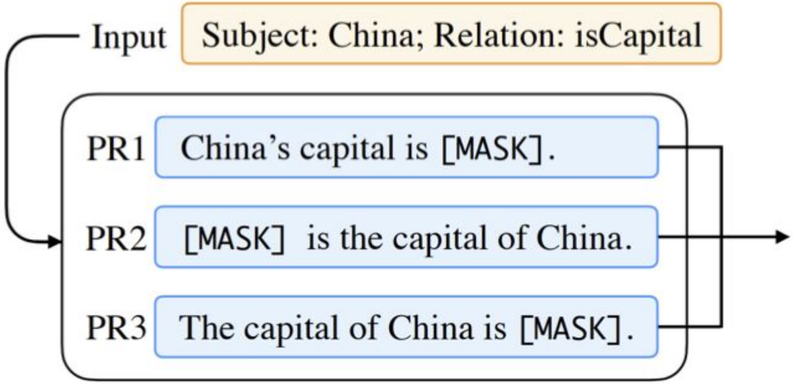
Expanding the paradigm



Prompt Ensemble and Augmentation

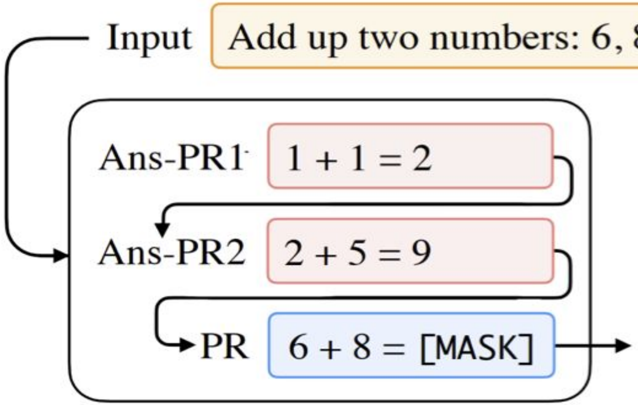
Ensembling:

Using multiple unanswered prompts for an input at inference time to make predictions.

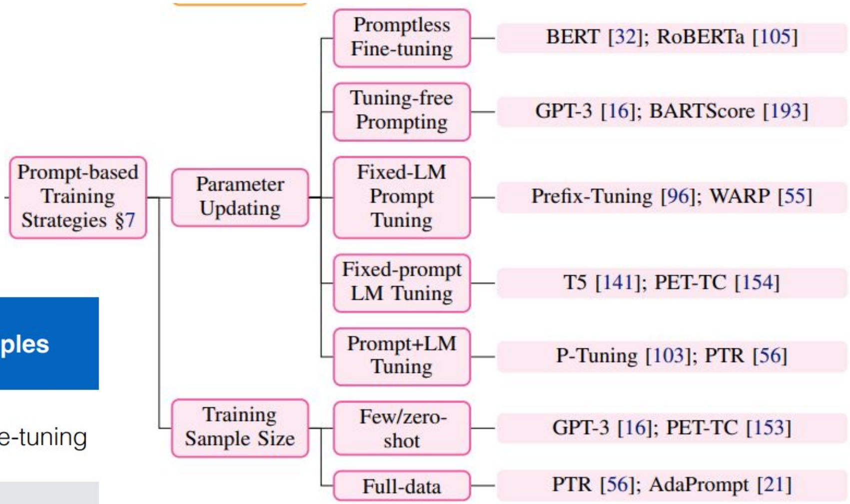


Augmentations:

Help the model answer the prompt that is currently being answered by additional answered prompts.

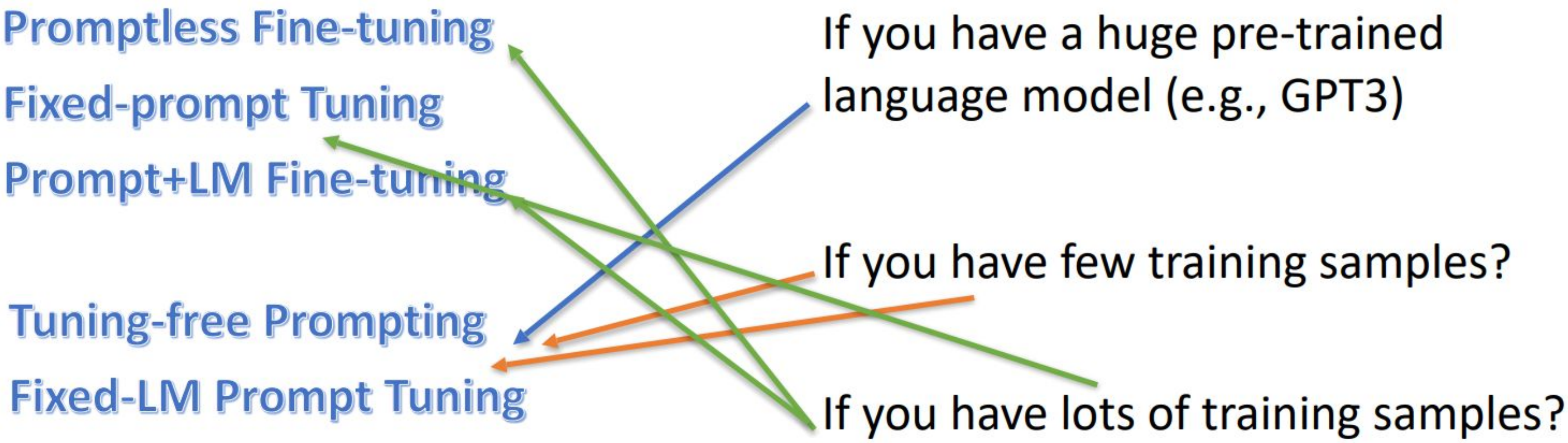


Prompt based Training strategy



Strategy	LM Params Tuned	Additional Prompt Params	Prompt Params Tuned	Examples
Promptless Fine-Tuning	Yes	N/A	N/A	BERT Fine-tuning
Tuning-free Prompting	No	No	N/A	GPT-3
Fixed-LM Prompt Tuning	No	Yes	Yes	Prefix Tuning
Fixed-prompt LM Tuning	Yes	No	N/A	PET
Prompt+LM Fine-tuning	Yes	Yes	Yes	PADA

Prompt based Training strategy



Strategy	LM Params Tuned	Additional Prompt Params	Prompt Params Tuned	Examples
Promptless Fine-Tuning	Yes	N/A	N/A	BERT Fine-tuning
Tuning-free Prompting	No	No	N/A	GPT-3
Fixed-LM Prompt Tuning	No	Yes	Yes	Prefix Tuning
Fixed-prompt LM Tuning	Yes	No	N/A	PET
Prompt+LM Fine-tuning	Yes	Yes	Yes	PADA

Discrete/Hard Prompt

- Natural language instruction and/or a few task demonstrations → output
- “Translate English to German:” That is good → Das is gut
- no gradient updates or fine-tuning

Problems:

- Requiring domain expertise/understanding of the model’s inner workings
- Performance still lags far behind SotA model tuning results
- Sub-optimal and sensitive
  - prompts that humans consider reasonable is not necessarily effective for language models
  - pre-trained language models are sensitive to the choice of prompts

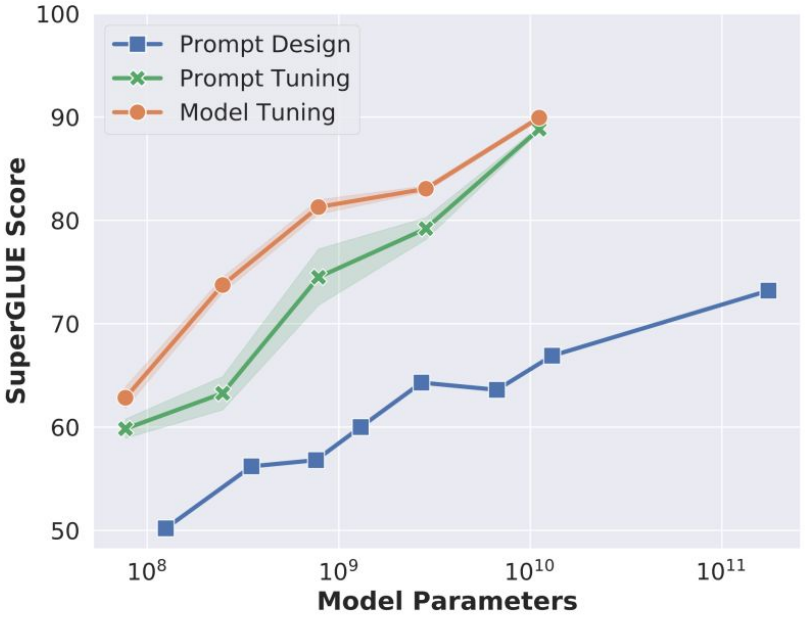
Discrete/Hard Prompt

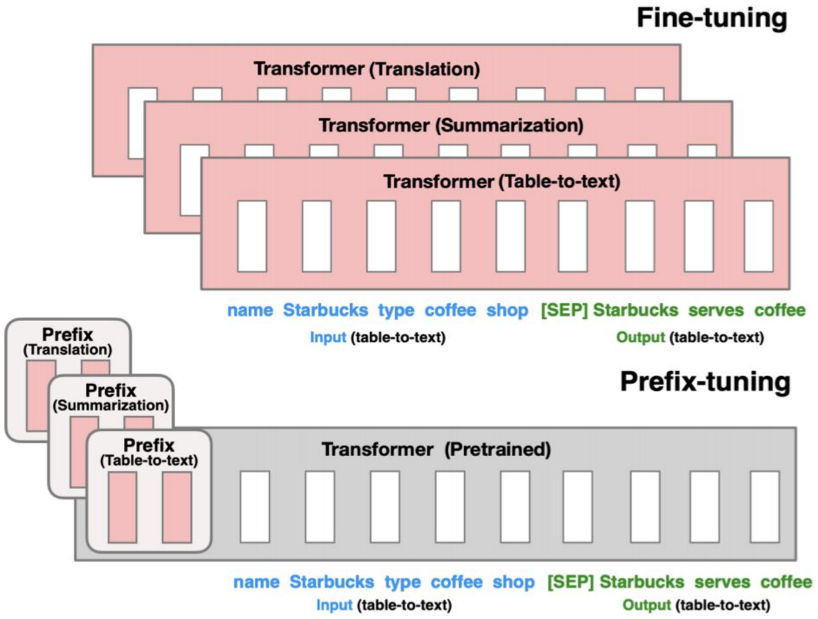
Prompt	P@1
[X] is located in [Y]. *(original)*	31.29
[X] is located in which country or state? [Y].	19.78
[X] is located in which country? [Y].	31.40
[X] is located in which country? In [Y].	51.08

Table 1. Case study on LAMA-TREx P17 with bert-base-cased. A single-word change in prompts could yield a drastic difference.

Continuous/Soft Prompt

- A sequence of additional task-specific tunable tokens prepended to the input text





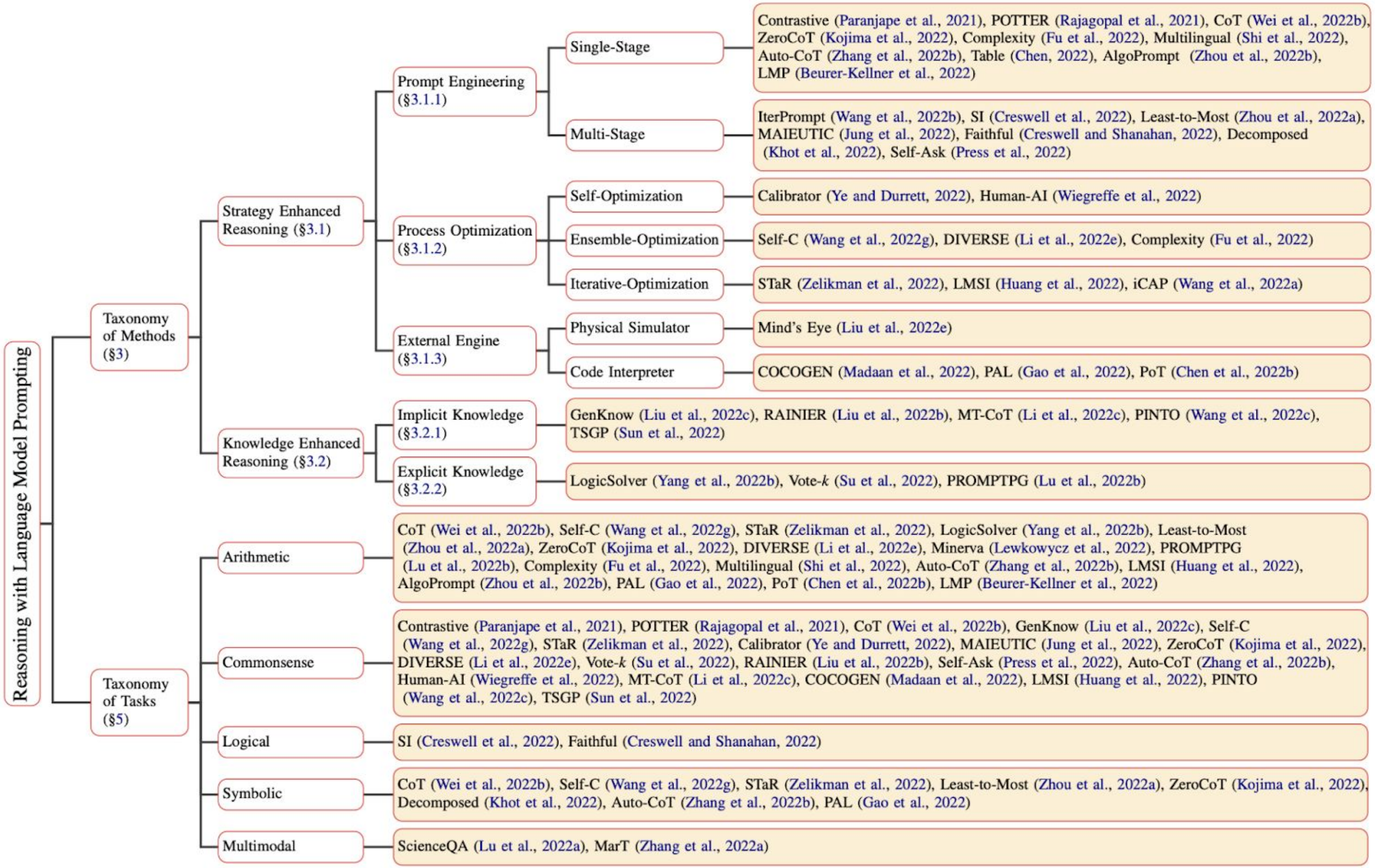
Reasoning with Large Language Prompting

Why to reason?

- Human thought is considered as the combination of two processes : thinking fast in System 1, which executes highly automated and largely effortless pattern recognition tasks, and thinking slow in System 2, which performs complex reasoning.
- Deep learning comprises one of the most successful artificial analogues of System 1, through its fast processing and pattern recognition.
- The challenge is to make end-to-end generic frameworks that are analogues to both System 1 and System 2.

Thinking, Fast and Slow is a 2011 book by psychologist Daniel Kahneman.

Reasoning with LLM prompting



Chain-of-thought Prompting

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ✗

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

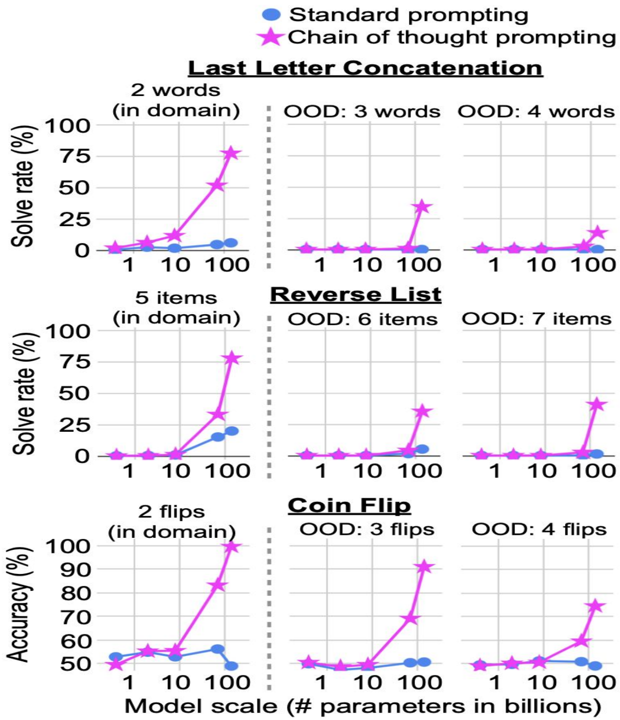
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

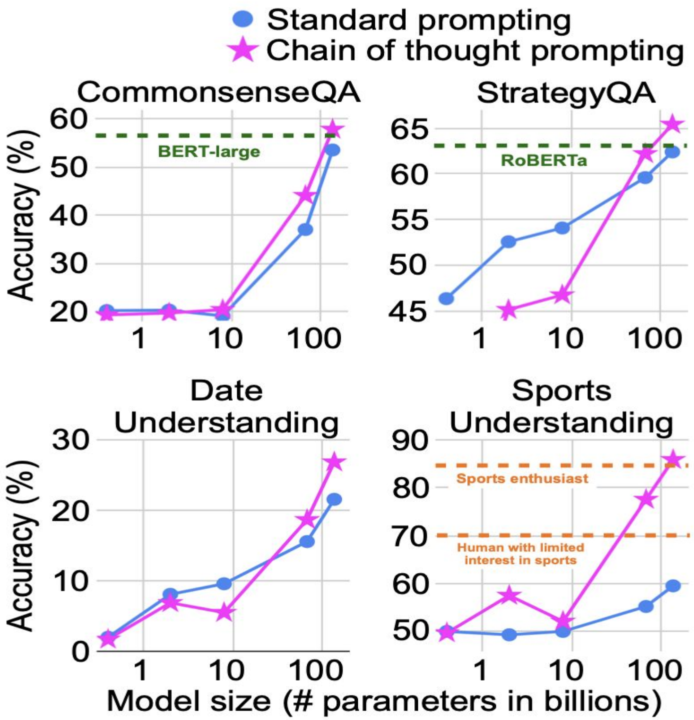
Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

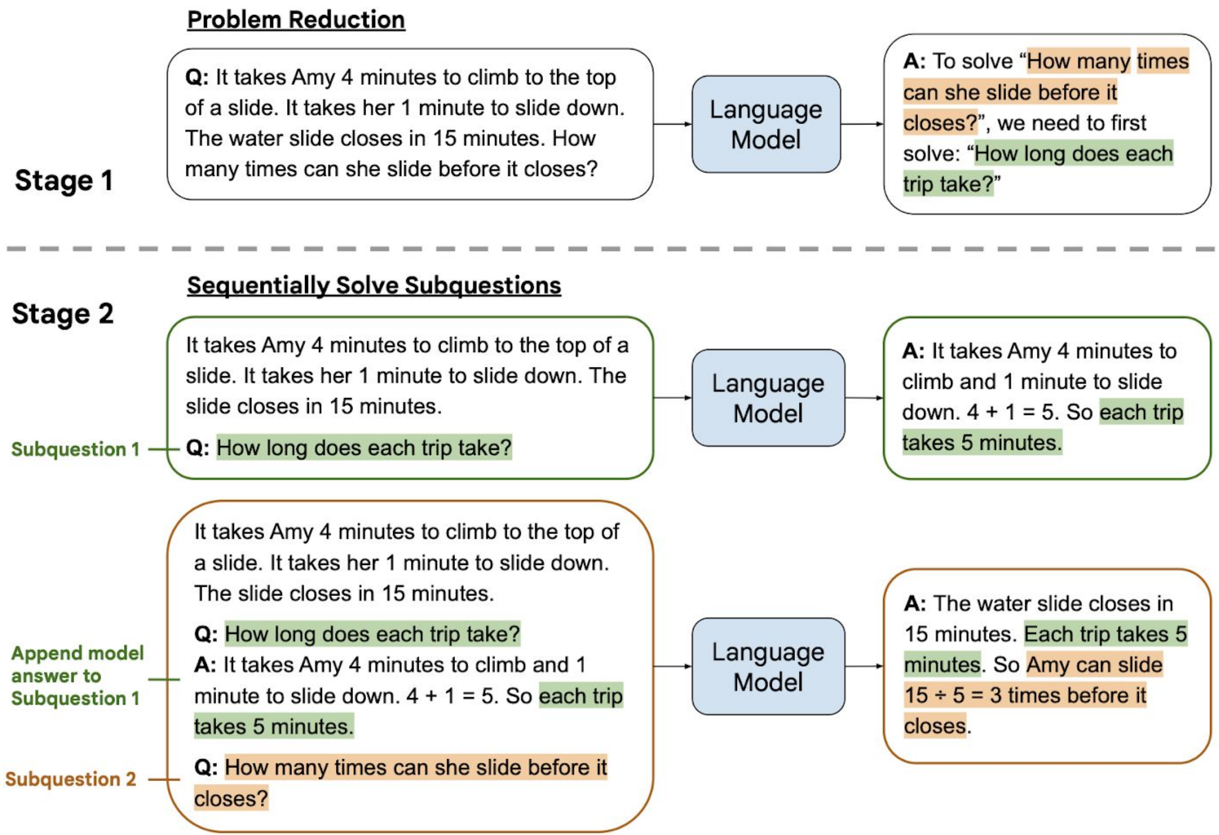
A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

Chain of thought prompting - Results





Least-to-Most Prompting



CoT and LtM Prompting

Chain-of-thought prompting	Least-to-most prompting (solving stage)
Q: “think, machine” A: The last letter of “think” is “k”. The last letter of “machine” is “e”. Concatenating “k”, “e” leads to “ke”. So, “think, machine” outputs “ke”. Q: “learning, reasoning, generalization” A: The last letter of “learning” is “g”. The last letter of “reasoning” is “g”. The last letter of “generalization” is “n”. Concatenating “g”, “g”, “n” leads to “ggn”. So, “learning, reasoning, generalization” outputs “ggn”.	Q: “think, machine” A: The last letter of “think” is “k”. The last letter of “machine” is “e”. Concatenating “k”, “e” leads to “ke”. So, “think, machine” outputs “ke”. Q: “think, machine, learning” A: “think, machine” outputs “ke”. The last letter of “learning” is “g”. Concatenating “ke”, “g” leads to “keg”. So, “think, machine, learning” outputs “keg”.

Ethics in NLP

PERFORMANCE MARKETING

Online Ads for High-Paying Jobs Are Targeting Men More Than Women

New study uncovers gender bias

Machine Bias

There's software used across the country to predict future criminals. And it's biased against blacks.

Amazon's Secret AI Hiring Tool Reportedly 'Penalized' Resumes With the Word 'Women's'

Rhett Jones
Yesterday 10:32am • Filed to: ALGORITHMS

Facebook, Citing Societal Concerns, Plans to Shut Down Facial Recognition System

Saying it wants "to find the right balance" with the technology, the social network will delete the face scan data of more than one billion users.

Jul 1, 2015, 01:42pm EDT

Google Photos Tags Two African-Americans As Gorillas Through Facial Recognition Software

Maggie Zhang Forbes Staff
Tech
I write about technology, innovation, and startups.

y and Greg More Employable than and Jamal? A Field Experiment on Labor Market Discrimination

Marianne Bertrand & Sendhil Mullainathan

Revealed: the software that studies your Facebook friends to predict who may commit a crime

- **Bias in statistics and ML**
    - Bias of an estimator: Difference between the predictions and the true values that we are trying to predict
    - The "bias" term b (e.g., $y = mx + b$)

- In a **Bayesian framework**, the prior P(X) serves as a bias: the expectation or base-rate we should have for something before we see any further evidence

- **Bias in Social Context:** Bias refers to being in favour or against/ preference or prejudice towards certain individuals, groups or communities based on their social identity (i.e., race, gender, religion etc.)
    - It reduces the time to take a decision.
    - Bias is an individual preference.
    - It can be either positive or negative.
    - Example : You hire an Asian for a job that has also an equally qualified black applicant. Reason: you think blacks are not as smart as Asians,this is a bias

How biases get into AI models ?

- Through Data
  - Bias present in annotations- Annotator Bias, e.g., *John's family has a doctor and a nurse; they are Jack and Jill;* Now coreference annotation may link doctor and *Jack* and nurse and *Jill*
  - Data Sampling Bias (choose 'convenient' data); e.g., t*railer of a movie shows only the 'attractive' snippets for marketability*
  - Representation Bias through embeddings (word2vec relies on context and distributional similarity; consequently cosine_similarity('doctor', 'male') > cosine_similarity ('doctor', female'))

- Through Models
  - bias in core algorithms/models lead to biased outputs
  - Loss function can have bias
  - bias in data + ML models leads to bias amplification

Leverage prompting for model debiasing

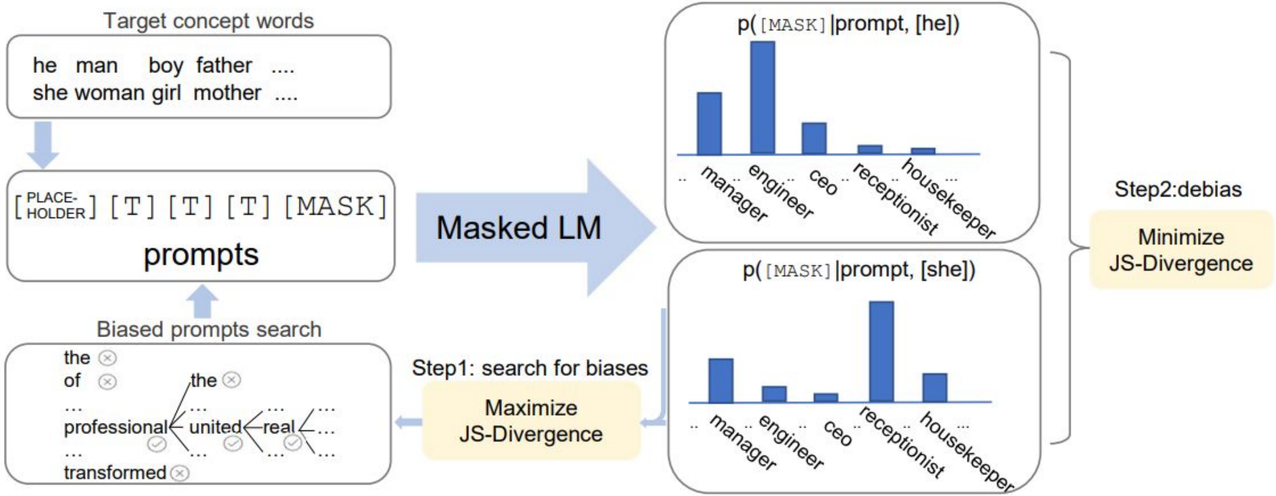


Figure 1: The Auto-Debias framework. In the first stage, our approach searches for the *biased prompts* such that the cloze-style completions (i.e., masked token prediction) have the highest disagreement in generating stereotype words. In the second stage, the language model is fine-tuned by minimizing the disagreement between the distributions of the cloze-style completions.

$$x_{\text{prompt}}(\text{she}) = \text{she has a job as } \texttt{[MASK]}.$$

$$p(\texttt{[MASK]} = v|\mathcal{M}, x_{\text{prompt}}(c)) = \frac{exp(\mathcal{M}_{\texttt{[MASK]}}(v|x_{\text{prompt}}(c)))}{\sum_{v' \in \mathcal{V}} exp(\mathcal{M}_{\texttt{[MASK]}}(v'|x_{\text{prompt}}(c)))}$$

References

[1] Amatriain, X.. "Transformer models: an introduction and catalog." *ArXiv* abs/2302.07730 (2023): n. Pag.

[2] Liu, Pengfei et al. "Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing." *ACM Computing Surveys* 55 (2021): 1 - 35.

[3] Brown, Tom B. et al. "Language Models are Few-Shot Learners." *ArXiv* abs/2005.14165 (2020): n. pag.

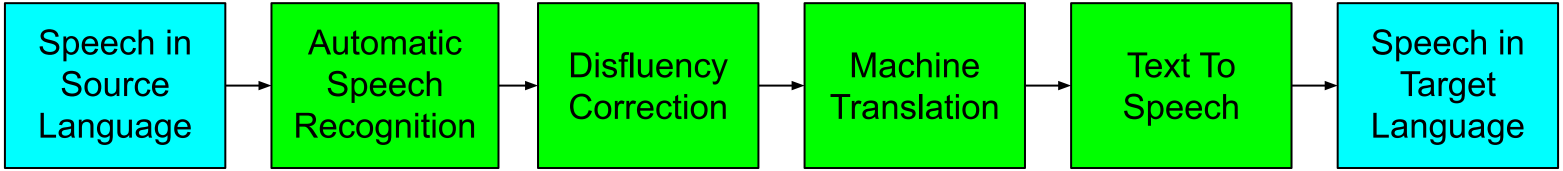
[4] https://people.cs.umass.edu/~miyyer/cs685_f22/slides/prompt_learning.pdf

[5] https://phontron.com/class/anlp2022/assets/slides/anlp-09-prompting.pdf

Speech-to-Speech Machine Translation

Problem Statement

- Speech-To-Speech Machine Translation (SSMT) is an automated process of converting speech in source language to speech in target language
- Two approaches -
  - Cascaded SSMT
  - End-to-end SSMT
- Components of a cascaded SSMT system:



Motivation

- Machine Translation is a well established field in NLP with early research dating back to 1950s
- Development of Deep Learning systems has resulted in high quality translation systems well supported by global efforts of generating parallel corpus in various languages
- Speech based input and output systems have capabilities to reach people around the world
- Speech-To-Speech Machine Translation systems become essential to connect millions of citizens especially in linguistically diverse countries like India

Automatic Speech Recognition

Goal

- ASR refers to the task of converting speech in source language to text in the same language
- Deep Learning based ASR systems have achieved SOTA performance in many languages and domains
- Performance of these systems is insufficient for Indian English due to nature of speech and dialect
- Our aim is to create an excellent quality transcription system for Indian English and Education Domain

Technique (1/3)

We benchmark three popular approaches in ASR -

1. Facebook's wav2vec 2.0
   - Pretraining CNN Encoders & Transformer on unlabelled speech data using self supervision
   - Finetuning on domain specific data in specific languages to achieve low word error rate

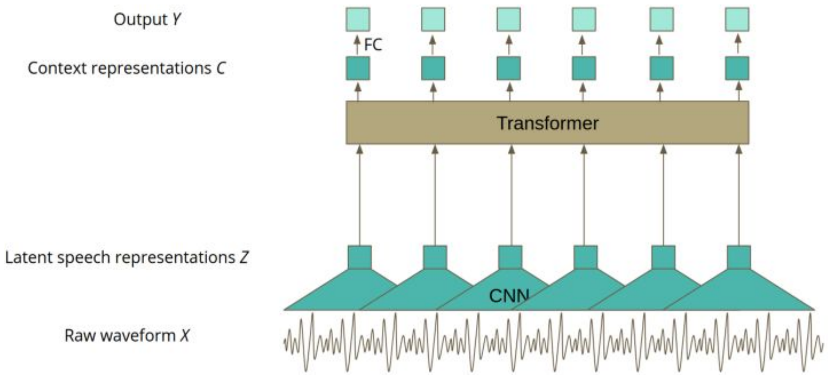


Fig 1: Architecture of wav2vec 2.0 [1]

Technique (2/3)

2. Vakyansh CLSRIL-23 ASR System:
   - Similar to wav2vec 2.0 with pre-training on 23 Indian languages followed by language specific finetuning
   - Achieves the best reported performance on ASR for Indian languages like Hindi, Marathi and Tamil

3. Open AI's Whisper ASR System:
   - Addresses the problems of wav2vec which creates more data centric models without attention to robustness

Technique (3/3)

- Converts audio into mel spectrograms before passing into Conv feature encoders
- Dataset used is **680K** hours of labelled data from various sources and diverse domains

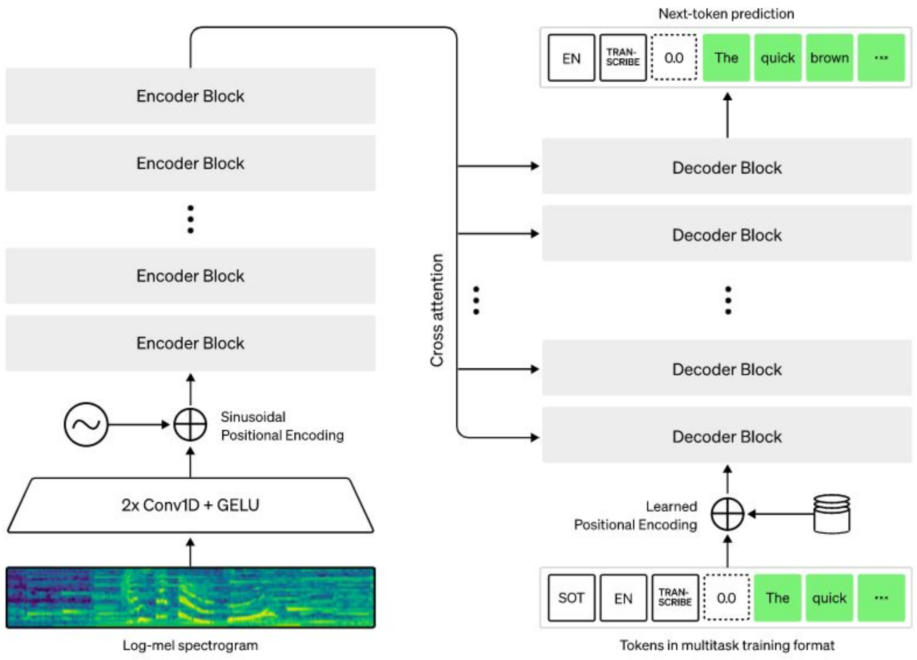


Fig 2: Architecture of Whisper ASR System [2]

Data

<u>Dataset:</u> NPTEL English Lectures (Only Test Set)

<u>Dataset Description:</u>

- No of hours = 1335.74
- Avg duration of clip = 7.69s
- Avg no of words per utterance = 17.33

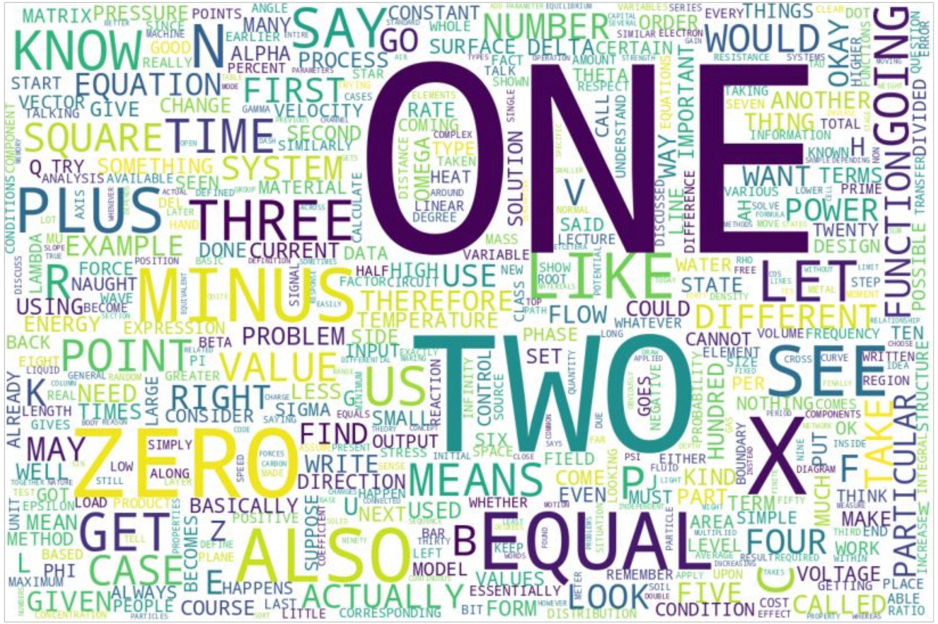


Fig 3: Word Cloud of corpus
(Excluding stop words)

Results

- We benchmark three techniques & finetune our current system on NPTEL corpus to compare Word error rate

Model name	Test WER
Wav2vec 2.0 (no finetuning)	49.2%
Vakyansh CLSRIL-23 (no finetuning)	32.8%
Open AI's Whisper ASR model	**28.2%**
Vakyansh CLSRIL-23 (with finetuning)	**28.4%**

Case Study 1

Sample		Observations - Whisper generates a random word in its transcription (common occurrence) but is able to hear the slightest of sounds at the end of the speech.
Generated Transcript (Finetuned Vakyansh)	that is you apply an input what kind of a transistor is this n p n not	
Generated Transcript (Whisper)	that is you apply your input what kind of a transistor is this N p n not p n.	
Reference Transcript	that is you apply an input what kind of a transistor is this npn not pn	

Sample		Observations - Whisper is able to catch the slightest of pronunciations for the first word. The speaker has a ambiguous pronunciation of the last word that affects the output. Word boundary detection is a problem for both models (common occurrence)
Generated Transcript (Finetuned Vakyansh)	see how we may describe culture bil this courses around these topics right by borrowing	
Generated Transcript (Whisper)	We see how we may describe culture, build this courses around these topics by borrowing	
Reference Transcript	we see how we may describe culture build discourses around this topics right by brewing	

Disfluency Correction

Introduction

- Conversational speech is spontaneous wherein the speaker is thinking about the content as they speak.
- Disfluencies are a set of words that occur in conversational speech that do not add any semantic meaning to the sentence.
- Speakers often use filler words, repeat fluent phrases or suddenly change content to make corrections in speech

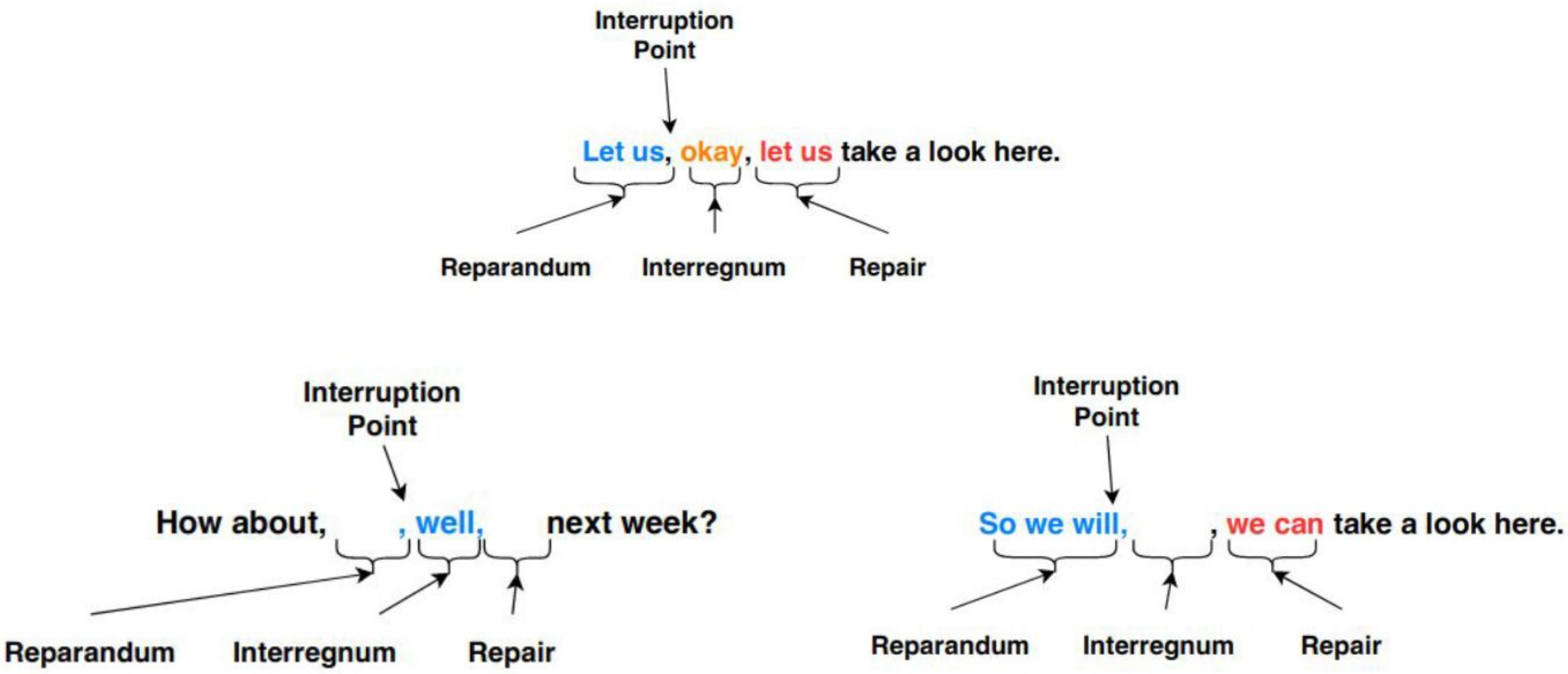
Example: **Well, this is** this is **you know** a good plan.

Types of Disfluencies

- There are 6 types of disfluencies we study in this thesis-

Type	Example
Filler	What are you **uh** doing tomorrow?
Interjection	**Oops**, I forgot to add your name under reservations.
Discourse Marker	**Well**, we don’t have to do it that way.
Repetition or Correction	**Let’s start**, let’s start the exam now.
False Start	**We’ll never find a day** what about next month?
Edit	We need **two tickets, I’m sorry**, three tickets to Delhi

Disfluency: Surface Structure



Disfluency Correction as Sequence Tagging

Disfluent: this this is a a big problem

Fluent: this is a big problem

Disfluent: this is this is a big problem

Fluent: this is a big problem

Tags: 1 1 0 0 0 0 0

- Fluent sentence is a subset of the complete sentence
- 0 corresponds to fluent and 1 corresponds to disfluent

Disfluency Correction in Indian Languages

Goal (1/2)

- Transcribing speech and annotating the disfluencies is a tedious task with an average of 5 min/sentence
- This makes creating a large DC corpus for Indian languages extremely challenging
- To achieve high quality results, we use a combination of labelled, unlabelled and pseudo labelled data from English and Indian languages

Goal (2/2)

- Disfluencies are words that are part of spoken utterances but do not add meaning to the sentence.
- We study disfluencies through 2 sources: Conversational Speech and Speech Impairments like Stuttering

Type	Example
Conversational	Well, you know, this is a good plan.
Stuttering	Um it was quite fu funny

Table 1: Examples and surface structure of disfluent utterances in conversational speech and stuttering. Red - Reparandum, Blue - Interregnum, Orange - Repair

Zero-shot Baseline and current SOTA

- The current SOTA for Indian languages DC is defined by Kundu et al. (2022) which uses a **Multilingual Transformer** architecture for token classification
- The model is trained on English data from the Switchboard corpus and synthetically generated data in Indian languages like Hindi, Marathi & Bengali
- Synthetic data was generated using rule based techniques for creating data in different disfluency types - a method that is clearly **not scalable**

Our Few Shot Approach

Three main components -

1. MuRIL Encoder - Generates feature vector (H_{real}) from real data
2. Generator: Creates hidden representations (H_{fake}) from gaussian random to mimic H_{real} & fool discriminator
3. Discriminator: If labeled data, Use H_{real} to classify disfluent/fluent tokens

   If unlabeled data, determine whether H_{fake} or H_{real} comes from a real distribution

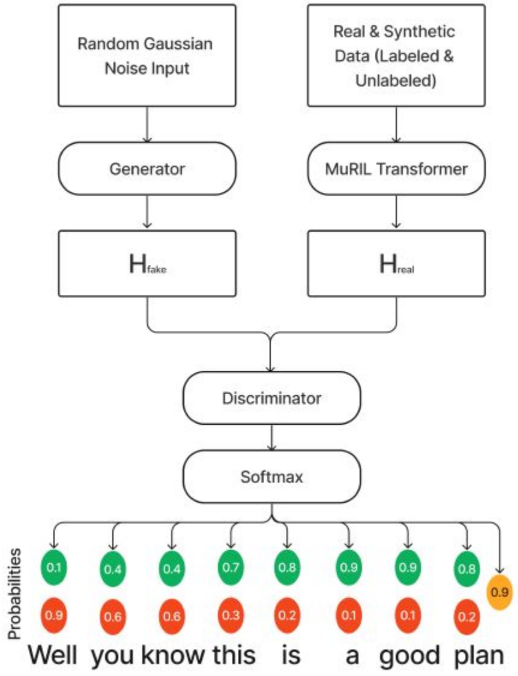


Figure 1: Architecture of the Seq-GAN-BERT model; Green nodes - Fluent class probabilities, Red nodes - Disfluent class probabilities, Orange node - Real (1) or Fake (0) probabilities

Code snippets (1/2)

```python
class Generator(nn.Module):
    def __init__(self, noise_size=100, hidden_size=512, dropout_rate=0.1):
        super(Generator, self).__init__()
        decoder_layer = nn.TransformerDecoderLayer(d_model=hidden_size, nhead=8)
        self.transformer_decoder = nn.TransformerDecoder(decoder_layer, num_layers=6)

    def forward(self, noise, memory):
        return self.transformer_decoder(noise, memory)


class Discriminator(nn.Module):
    def __init__(self, input_size=512, hidden_sizes=[512], num_labels=2, dropout_rate=0.1):
        super(Discriminator, self).__init__()
        self.input_dropout = nn.Dropout(p=dropout_rate)
        layers = []
        hidden_sizes = [input_size] + hidden_sizes
        for i in range(len(hidden_sizes)-1):
            layers.extend([nn.Linear(hidden_sizes[i], hidden_sizes[i+1]), nn.LeakyReLU(0.2, inplace=True), nn.Dropout(dropout_rate)])

        self.layers = nn.Sequential(*layers)
        self.logit = nn.Linear(hidden_sizes[-1],num_labels+1) # +1 for the probability of this sample being fake/real.
        self.softmax = nn.Softmax(dim=-1)

    def forward(self, input_rep):
        input_rep = self.input_dropout(input_rep)
        last_rep = self.layers(input_rep)
        logits = self.logit(last_rep)
        probs = self.softmax(logits)
        return last_rep, logits, probs
```

```python
model_outputs = transformer(b_input_ids, attention_mask=b_input_mask)
hidden_states = model_outputs[0]


# Generator using Transformer Decoder
noise = torch.randn(max_seq_length, real_batch_size, hidden_size).to(device)
memory = torch.randn(max_seq_length, real_batch_size, hidden_size).to(device)
gen_rep = generator(noise, memory).permute(1, 0, 2)

# Generate the output of the Discriminator for real and fake data.
discriminator_input = torch.cat([hidden_states, gen_rep], dim=0)

# Then, we select the output of the disciminator
features, logits, probs = discriminator(discriminator_input)

# Finally, we separate the discriminator's output for the real and fake data
features_list = torch.split(features, real_batch_size)
D_real_features = features_list[0]
D_fake_features = features_list[1]

logits_list = torch.split(logits, real_batch_size)
D_real_logits = logits_list[0]
D_fake_logits = logits_list[1]

probs_list = torch.split(probs, real_batch_size)
D_real_probs = probs_list[0]
D_fake_probs = probs_list[1]
```

Code snippets (2/2)

```
# Generator's LOSS estimation
g_loss_d = -1 * torch.mean(torch.log(1 - D_fake_probs[:, :, -1] + epsilon))
g_feat_reg = torch.mean(torch.pow(torch.mean(D_real_features, dim=0) - torch.mean(D_fake_features, dim=0), 2))
g_loss = g_loss_d + g_feat_reg

# Discriminator's LOSS estimation
logits = D_real_logits[:, :, 0:-1]
log_probs = F.log_softmax(logits, dim=-1)
label2one_hot = torch.nn.functional.one_hot(b_labels01, len(label_list))
per_example_loss = -torch.sum(label2one_hot * log_probs, dim=(-2, -1))
per_example_loss = torch.masked_select(per_example_loss, b_label_mask.to(device))
labeled_example_count = per_example_loss.type(torch.float32).numel()
```

```
# It may be the case that a batch does not contain labeled examples, so the "supervised loss" in this case is not evaluated
if labeled_example_count == 0:
    D_L_Supervised = 0
else:
    D_L_Supervised = torch.div(torch.sum(per_example_loss.to(device)), labeled_example_count)

D_L_unsupervised1U = -1 * torch.mean(torch.log(1 - D_real_probs[:, :, -1] + epsilon))
D_L_unsupervised2U = -1 * torch.mean(torch.log(D_fake_probs[:, :, -1] + epsilon))
d_loss = D_L_Supervised + D_L_unsupervised1U + D_L_unsupervised2U
```

Results (1/2)

- Results on Indian languages DC

Lang	Model	P	R	F1
Bn	Baseline I	93.06	62.18	74.55
	Baseline II	66.37	68.20	67.27
	Baseline III	84.00	78.93	81.39
	Our model	87.57	80.23	**83.74**
Hi	Baseline I	85.38	79.41	82.29
	Baseline II	82.99	81.33	82.15
	Baseline III	88.15	83.14	85.57
	Our model	89.83	86.51	**88.14**
Mr	Baseline I	87.39	61.26	72.03
	Baseline II	82.00	60.00	69.30
	Baseline III	84.21	64.21	72.86
	Our model	85.34	67.58	**75.43**

Table 2: Comparing the performance of baselines and our model on DC across Bengali (Bn), Hindi (Hi) and Marathi (Mr); Baseline I - Monolingual supervised training, Baseline II - Multilingual supervised training, Baseline III - Adversarial training without unlabeled data, Our model - Multilingual adversarial training with unlabeled data; P = Precision, R = Recall

- Stuttering DC in English

Model	P	R	F1
Baseline I	89.11	78.08	83.23
Baseline II	87.34	86.50	86.92
Baseline III	74.58	86.33	80.02
Baseline IV	85.76	84.17	84.96
Baseline V	86.21	84.82	85.51
Our model	87.26	88.10	**87.68**

Table 3: Comparing the baselines and our model for stuttering DC in English; Baseline I - Supervised training on gold standard dataset, Baseline II and III - Supervised training on gold standard dataset and DC data, Baseline IV and V - Adversarial training without unlabeled data, Our model - Multilingual Adversarial training with unlabeled data; P = Precision, R = Recall

Results (2/2)

- We also show the qualitative difference between zero shot and few shot model

- ZS model had a high precision but a very low recall. FS Model improved the recall without compromising precision significantly

Lang	Input	Transliteration	Gloss	Translation	ZS Output	FS Output
Bn	বিষয় স্যার বিষয়টা স্যার স্যার আমি একটু ভুল বললাম	biShaya syaara biShay-aTaa syaara syaara aami ekaTu bhula balalaama	subject sir the_matter sir sir I_am a_little wrong I_said	Subject Sir Subject Sir Sir I said a little wrong	বিষয়টা আমি একটু ভুল বললাম	বিষয়টা স্যার আমি একটু ভুল বললাম
Hi	तो यह है अ स्कुल	to yaha hai a skula	so it is a school	so this is uh school	यह है अ स्कुल	तो यह है स्कुल
Hi	बहत तेज चलाते थे और मैं अ क्या कहते है ह एनिमलस गिनता था रास्ते मैं	bahata teja chalaate the aura mai.m a kyaa ka-hate hai ha enimalasa ginataa thaa raaste mai.m	a_lot quick drive were and I a what say is h an-imals count was way I	Used to drive very fast and I used to count the animals on the way	बहत तेज चलाते थे और मैं अ क्या कहते है ह एनिमलस गिनता था रास्ते मैं	बहत तेज चलाते थे और मैं ह एनिमलस गिनता था रास्ते मैं
Mr	मी आज अं फुलांचे जे प्रदर्शन पाहिले त्यात व्हर्टीकल गार्डनची संकल्पना पाहायला मिळाली	mii aaja a.m phu-laa.mche je pradarshana paahile tyaata vharTiikala gaarDanachii sa.mkalpanaa paahaayalaa mildaalii	I today uh of_flowers j exhibi-tion saw in_it vertical of_the_garden concept to_see re-ceived	The concept of vertical garden was seen in the exhibition I saw today	आज अं फुलांचे जे प्रदर्शन पाहिले त्यात व्हर्टीकल गार्डनची संकल्पना पाहायला मिळाली	मी आज फुलांचे जे प्रदर्शन पाहिले त्यात व्हर्टीकल गार्डनची संकल्पना पाहायला मिळाली
Mr	देशातील प्रत्येक शहरात प्रत्येक गावात ही स्वच्छता मोहीम सुरू आहे	deshaatiila pratyeka shaharaata pratyeka gaavaata hii svachChataa mohiima suruu aahe	in_the_country each in_the_city each in_the_village this cleanli-ness cam-paign con-tinue is	This cleanli-ness drive is going on in every city in every village of the coun-try	देशातील प्रत्येक गावात ही स्वच्छता मोहीम सुरू आहे	देशातील प्रत्येक गावात ही स्वच्छता मोहीम सुरू आहे

Table 5: Comparison between the output of the zero-shot and few-shot model. The Few-Shot model provides better inference in most cases; Bn - Bengali, Hi - Hindi, Mr - Marathi, ZS - Zero Shot, FS - Few Shot

Contributions

- Improving the state-of-the-art in DC in Bengali, Hindi and Marathi by 9.19, 5.85 and 3.40 points in F1 scores
- Creating an open-source stuttering English DC corpus comprising 250 parallel sentences
- Demonstrating that our adversarial DC model can be used for textual stuttering correction with high accuracy (87.68 F1 score)

An interesting problem in Indian Languages: Reduplication vs Repetition

- Reduplication refers to the phenomenon of repeating words for greater emphasis of certain phrases.
- Repetition is a disfluency type where words are repeated in conversations as disfluent words
- Reduplication is not a disfluency whereas repetition is

Example	Explanation
मुझे लाल लाल टमाटर चाहिए	Here “लाल लाल” is a phrase with one word repeated but it is not a disfluency since the word is repeated for greater emphasis on the redness of the tomatoes
तुम कहाँ कहाँ गए थे?	Similarly, “कहाँ कहाँ” is an example of reduplication signifying emphasis on the places the person went

Low Resource Text to Speech

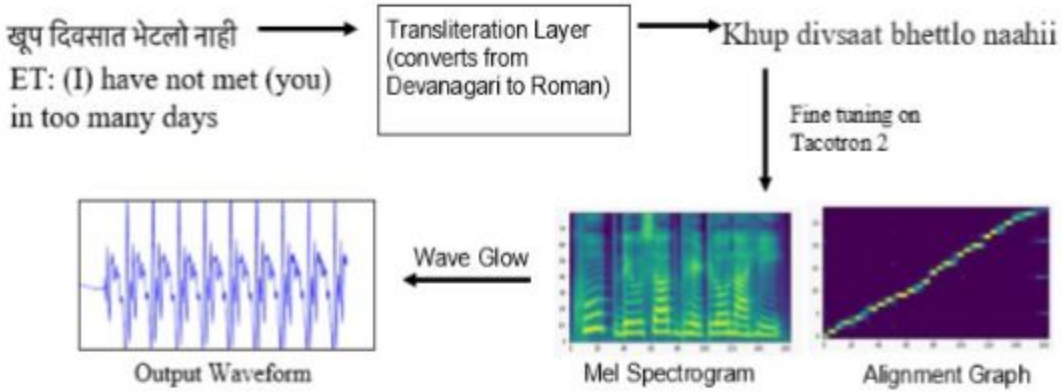
Goal

- Speech synthesis data must be clear of background noise and pronunciations must be consistent
- Such datasets are limited in English and more so in Indian languages.
- Create a high quality speech synthesis system using a novel transliteration strategy for domain transfer to high resource languages like English
- We experiment our transliteration approach on auto-regressive and non-autoregressive models

Technique (1/2)

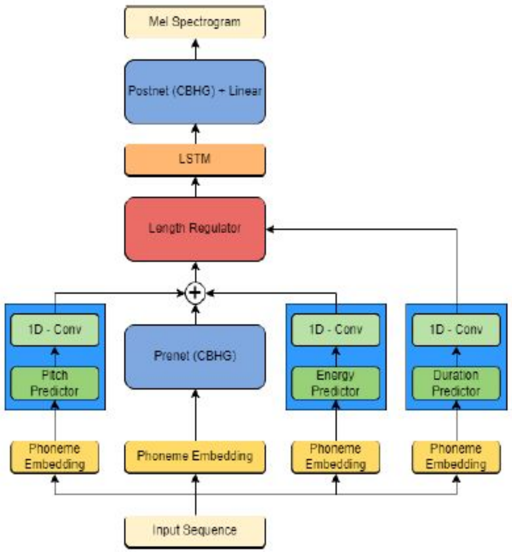
- TTS consists of two parts - Spectrogram generator and Vocoder
- We explore two models for the spectrogram generator-

1. Tacotron 2
   - Encoder-Decoder structure with convolutional layers, batch normalization & ReLU layers
   - Autoregressive approach



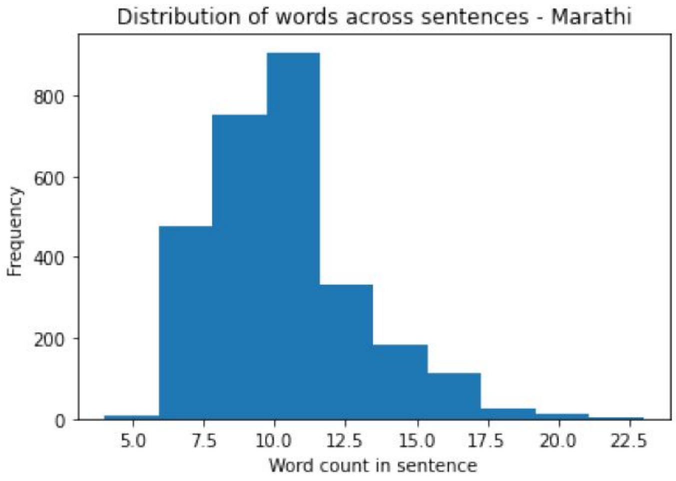
Technique (2/2)

2. Forward Tacotron
   - Non-autoregressive model which can predict mel spectrograms in a single forward pass
   - 3 Seq2Seq models are trained for predicting duration, pitch & energy for each input token
   - Length regulator is trained separately to expand on input sequence to generate mel spectrograms in one pass



Data

- Use Marathi Text-Speech data from Indic-TTS Corpus
  - Consisted of 4.82 hrs of data; Mean duration=7.09s
  - Clear pronunciations & negligible background noise



Results

Method 1: Tacotron2 + Waveglow vocoder

- Mean Opinion Score = 4.53 out of 5 (Survey conducted as a part of evaluation, 118 people included)

Method 2: Forward Tacotron + Waveglow vocoder

- Training was performed with Eng checkpoints until Mel generation loss & duration prediction loss saturated
- Mean Opinion Score = 4.64 out of 5 (Survey conducted as a part of evaluation, 118 people included)

Case Study 2

Input Text	मिहीर चांगला माणूस होता	Observations - The phoneme ‘cã’ is pronounced much better in Forward Tacotron compared to tacotron 2.
English Translation	Mihir was a good man	
Audio Generated (Tacotron 2)		
Audio Generated (Forward Tacotron)		
Input Text	मी बाजारात जातो.	Observations - Pronunciations are very similar but forward tacotron has a higher pitch and more accurate intonation
English Translation	I go to the market.	
Audio Generated (Tacotron 2)		
Audio Generated (Forward Tacotron)		

Future Work (1/2)

- ASR
  - Training Whisper model on labelled Indian languages ASR
  - Dialect adaptation and applications in Closed Captioning
- DC
  - Collecting more data and expanding to other Indian languages
  - Work on reduplication vs repetition

Future Work (2/2)

- TTS
  - Integrating sentiment in speech synthesis to generate more human like speech
  - Using learnable grapheme to phoneme modules to expand our work in non phonetic languages

Thank You!

Automatic Post-Editing (APE) and Quality Estimation

Guide : Prof. Pushpak Bhattacharyya

Introduction

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:
  - To reduce the human effort in post-editing phase
  - Black-box scenario: To further improve translations by identifying and correcting recurring MT errors

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.
  - Translation (from MT): लसीकरणानंतरही लोकांना कोविड - 19 **मिळू** शकतो .
  - Human post-edited version: लसीकरणानंतरही लोकांना कोविड - 19 **होऊ** शकतो .

- Input: MT translation (and Source sentence), 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

APE Paradigms (1/2)

- APE task: a monolingual translation task
  - The same MT technology is 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

Summary: WMT APE Shared Tasks

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

- Quality of post-edits: post-edits from professional post-editors
- Type of underlying MT system and quality of translations
- Technology Development: Utilization of more and more data
- The difficulty of the APE task: inversely proportional to quality of machine translation system
- Problem of Over-correction
- Uncertainty about effectiveness of current neural approaches

Terminologies

- SRC (source): source language sentence
- MT_OP (translation): 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: Output generated by the APE system

- Synthetic APE Data: (SRC, MT_OP, MT_REF)
- Real APE Data: (SRC, MT_OP, PE_REF)

WMT22 English-Marathi APE Shared Task Submission

Problem Statement and Data

- To develop a robust English-Marathi APE system using the data shared in the WMT22 APE Shared Task.

- Triplets: source_sentence (SRC), MT_output (MT_OP), Human post-edited version of MT (PE_REF)
- Synthetic Data:
  - MT Parallel corpus (SRC, MT_Ref) –> APE data (SRC, MT_OP, MT_Ref)

- Dataset:
  - Synthetic APE data: around 2M triplets
  - Real APE data: 18K triplets
  - Validation, Test data: 1K triplets each

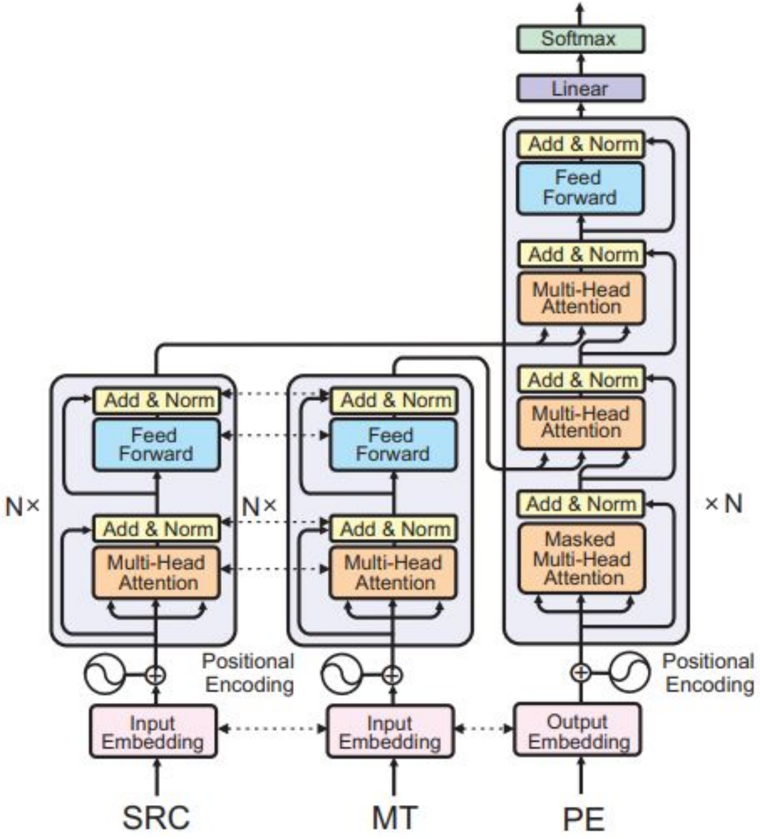
Features

- Model Architecture: Two-encoders single-decoder model.
  - No vocabulary overlap between English and Marathi data, different scripts.
- Use of LaBSE-based data filtering:
  - We observed that the quality of the synthetic data was not high.
- Data Augmentation using Phrase-level Triplet Generation:
  - Phrase-level triplets may help the APE system learn the phrase-level alignments which can help to identify errors and correct only certain segments of the sequence.
- Training the Model using Curriculum Training Strategy (CTS):
  - Our CTS considers in-domain and out-domain triplets and their TER scores. It allows the model to learn more error patterns and also forces it to not make more edits.
- Use of Sentence-QE as a final output selector:
  - Allows to handle cases of over-correction

Approach (1/2)

- Data Pre-processing:
    - LaBSE-based Filtering to filter low-quality triplets
    - Form subsets of the data based on domain

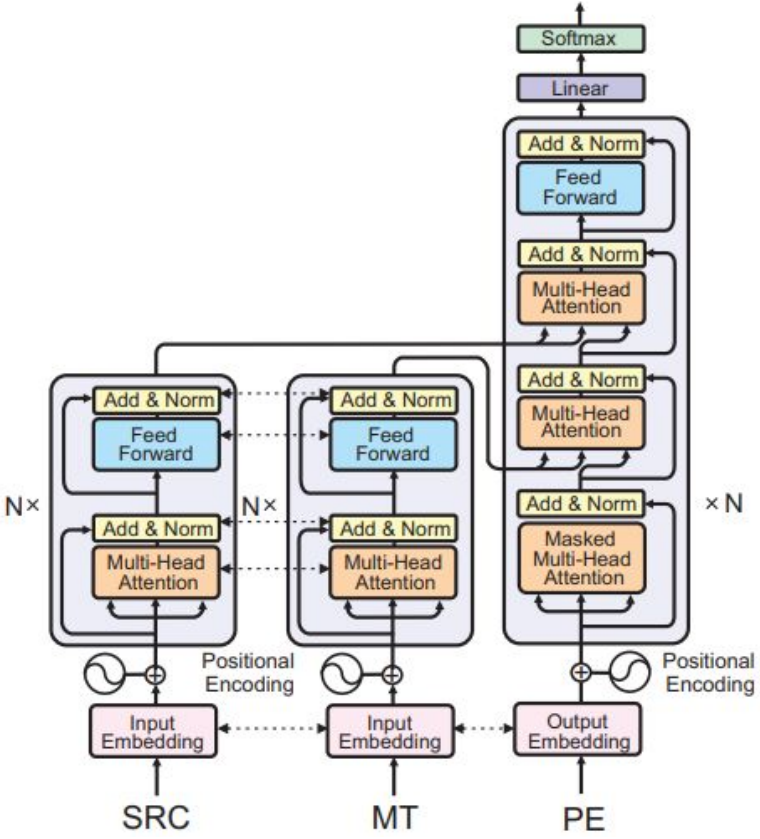
- Data Augmentation:
    - Phrase-level APE Triplet Generation:
        - Extract SRC-MT and SRC-PE phrase tables
        - Generate MT-PE triplets
    - External MT triplets: using mT5-fine-tuned NMT model



Approach (2/2)

- Training (using Curriculum Training Strategy):
  Train the model on
  - Step 1: MT task
  - Step 2: APE task using out-of-domain synthetic APE triplets
  - Step 3: APE task using in-domain high TER synthetic APE triplets
  - Step 4: APE task using low TER synthetic APE data augmented with External MT candidates (in-domain)
  - Step 5: APE task using the real-APE data (Fine-tuning)

- Selection of Final Output:
  - Sentence-level Quality Estimation to select the final output: the original translation or the APE hypothesis



Results

- Results on the WMT22 Development Set [1]:

System	TER↓	BLEU↑
Do Nothing (Baseline)	22.93	64.51
+ CTS-based Training and External MT	20.08	67.39
+ LaBSE-based Data Filtering and in-domain training data	19.73	67.86
+ Phrase-level APE triplets	19.39	68.35
+ Sentence-level QE	**19.01**	**68.87**

- Results on the WMT22 Test Set [2]:

- % of sentences:
  - Modified: 45.2
  - Improved: 63.5
  - Deteriorated: 27.9

		TER	BLEU
en-mr	IITB_APE_QE_combined_PRIMARY.tsv	**16.79**	**72.92**
	LUL_HyperAug_Adaptor_CONTRASTIVE	**19.06**	**69.96**
	LUL_HyperAug_Finetune_PRIMARY	**19.36**	**69.66**
	baseline (MT)	20.28	67.55
	IIIT-Lucknow_adversia-machine-translation_PRIMARY.txt	57.14	23.43
	IIIT-Lucknow_adversia-machine-translation_CONTRASTIVE.txt	99.81	3.16

Qualitative Observations (1/2)

- Handling of deletion cases, Improvement in lexical choice:
  - Source: This will **contribute to** improvements in the living standards of the **underprivileged** population of the society.
  - MT: यामुळे समाजातील गरीब लोकांच्या राहणीमानात सुधारणा होईल.
  - APE_OP: हे समाजातील **वंचितांच्या** राहणीमानात सुधारणा करण्यास **हातभार** लावेल.

- Handling negation:
  - Source: PW-10/A was **not** her statement.
  - MT: पीडब्ल्यूडब्ल्यू 10/ए तिचे विधान मागे घेत होते.
  - APE_OP: पीडब्ल्यू -10/ए हे तिचे विधान **नव्हते**.

Qualitative Observations (2/2)

- Tense Modification:
  - En: With these directions, the petition stands disposed of.
  - MT_OP: या निर्देशांसह सहपत्रांची याचिका निकाली काढली जाऊ शकते.
  - APE_OP: या निर्देशांसह याचिका निकाली काढण्यात आली.

- Unnecessary Insertion:
  - En: Such a person is called a Shaheed.
  - MT_OP: अशा व्यक्तीला शहीद म्हणतात.
  - PE (Joint Encoding + CTS): अशा व्यक्तीला शहीद **धर्म** सांगण्याची **तातडीने**.
  - PE: अशा व्यक्तीला शहीद **धर्म सांगतात**.

Quality Estimation

MT Evaluation - Referenceless Translations

- **Quality Estimation** task:

  Score the translation quality given just source text and translated text.

- **Levels of QE:**
  - **Document level**
  - **Sentence level**
  - **Word level**

QE Research Motivation

- Effective Evaluation requires:
  - **Multiple reference translations**.
  - **Time** and **effort** by expert translators.
- **Transfer learning evaluation** for **low-resource languages**.
- **Word level QE** could help with **post-editing** efforts.

QE at Different Granularities

- **Word Level QE:** At the word level QE, each word and gap in the sentence in assigned an ‘OK’ or ‘BAD’ tag. The meaning of these tags is as follows:
  - Source sentence words: ‘BAD’ tag indicates a source sentence word leading to an incorrect translation in target sentence.
  - Target sentence words: ‘BAD’ tag indicates an incorrectly translated word in target sentence.
  - Target sentence gaps: ‘BAD’ tag indicates a missing word in target sentence.
- **Sentence Level QE:** A QE prediction score for each translated sentence pair.
- **Document Level QE:** A QE prediction score for the entire translated document pair.

MT Evaluation - QE Metrics

- **HTER (Human Translation Error Rate):**
  Ratio of **number of edits** (insertions, deletions, substitutions, shifts) to **reference sentence length**.
- **DA (Direct Assessment):**
  A **translation quality score** on a scale **0-100** by professional human translators. **Z-scores** of multiple evaluators considered.
- **HTER is unable** to capture **adequacy** properly. **Fluent** yet incorrect translations scored **highly** by SOTA QE systems.

Reference: Are we Estimating or Guesstimating Translation Quality? Shuo Sun, Francisco Guzmán, and Lucia Specia. 2020. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

QE - Examples

- Sentence Level Direct Assessment Score:
  - [En]The weather is good today. [Hn] आज मौसम ठीक है। — DA : 80
  - [En]The weather is good today. [Hn] आज का मौसम। — DA : 35
  - Multiple Annotators, mean Z-Score for all annotations taken into consideration for model predictions.
- Sentence Level Direct Assessment Score:
  - [En]The weather is good today. [Hn] कल मौसम है।
  - [En] Tags [OK OK OK BAD BAD]
  - [Hn] Tags [OK BAD OK OK BAD OK OK]

Problem Statement and Motivation

- Problem Statement:
  - Developing a robust QE model to perform Sentence-level and Word-level QE tasks.
- Motivation:
  - Sentence-level and Word-level QE tasks are related to each other.
  - Having separate models for word-level and sentence-level QE tasks can result in inconsistent outputs for the same inputs.

Contributions

- Showing that jointly training a model using Multi-Task Learning (MTL) for sentence and word-level QE tasks improves performance on both tasks. In a single-pair setting, we observe an improvement of up to 3.48% in Pearson's correlation (*r*) at the sentence-level and 7.17% in *F1-score* at the word-level.

- Showing that the MTL-based QE models are more consistent, on word-level and sentence-level QE tasks, for same inputs, as compared to the single-task learning-based QE models.

- To the best of our knowledge, we introduce a novel application of the Nash-MTL method to both tasks in Quality Estimation.

Approach (1/2)

Sentence-level QE loss:

$$\mathcal{L}_{da} = MSE\left(\mathbf{y}_{da}, \hat{\mathbf{y}}_{da}\right)$$

Where, '*da*' in L_{da} stands for '*direct assessment*.'

Word-level QE loss:

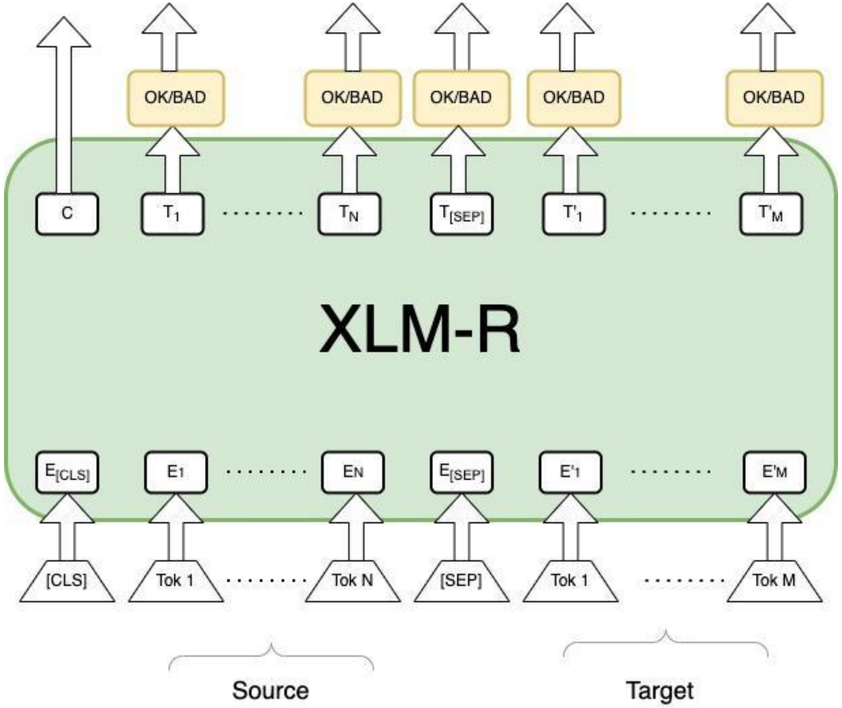
$$\mathcal{L}_{word} = -\sum_{i=1}^{2}\left(\mathbf{y}_{word} \odot \log(\hat{\mathbf{y}}_{word})\right)[i]$$

Where, $\odot$ denotes element-wise multiplication.
[i] retrieves *i*th item in the vector.

Linear Scalarization (**LS-MTL**):

$$\mathcal{L}_{MultiTransQuest} = \frac{\alpha\mathcal{L}_{da} + \beta\mathcal{L}_{word}}{\alpha + \beta}$$

where, α and β are kept at 1.



Approach (2/2)

Nash-MTL: The method arranges bargaining between weight update directions of each task.

Algorithm 1 Nash_MTL

Input: θ_0 - initial parameter vector, $\{l_i\}_{i=1}^{K}$ - differentiable loss functions, η - learning rate

Output: θ^T

for $t = 1,..., T$ **do**

  Compute task gradients $g_i^t = \nabla_{\theta(t-1)} l_i$

  Set $G^{(t)}$ the matrix with columns $g_i^{(t)}$

  Solve for $\alpha : (G^t)^T(G^t)\alpha = 1/\alpha$ to obtain α^t

  Update the parameters $\theta^{(t)} = \theta^{(t)} - \eta G^{(t)}\alpha^{(t)}$

end for

return θ^T

Datasets

- We use data released in the WMT20 and WMT22 QE shared tasks.

- Language Pairs and number of samples in Training set:
  - Low-resource: En-Mr (20K), Ne-En (7K), Si-En (7K)
  - Mid-resource: Et-En (7K), Ro-En (7K), Ru-En (7K)
  - High-resource: En-De (7K)

- Number of samples in:
  - Validation set: 1K for each language pair
  - Test set: 1K for each language pair

Experimental Settings

- We evaluated our approach in three experimental settings explained below:
  - **Single-Pair Setting**: We only use the data of one language pair for training and evaluation.
  - **Multi-Pair Setting**: We combine the data of all language pairs for training and evaluate on each language pair.
  - **Zero-Shot Setting**: We combine the data of all language pairs for training except the language pair on which we want to evaluate the model.

Results: Single-Pair Setting

LP	Word-Level					Sentence-Level				
	STL	LS-MTL	+/- %	Nash-MTL	+/- %	STL	LS-MTL	+/- %	Nash-MTL	+/- %
En-Mr	0.3930	0.4194	2.64%	**0.4662**	7.32%	0.5215	0.5563	3.48%	**0.5608**	3.93%
Ne-En	0.4852	0.5383	5.31%	**0.5435**	5.83%	0.7702	0.7921	2.19%	**0.8005**	3.03%
Si-En	0.6216	0.6556	3.40%	**0.6946**	7.30%	0.6402	0.6533	1.31%	**0.6791**	3.89%
Et-En	0.4254	0.4971	7.17%	**0.5100**	8.46%	0.7646	0.7905	2.59%	**0.7943**	2.97%
Ro-En	0.4446	0.4910	4.64%	**0.5273**	8.27%	0.8952	**0.8985***	0.33%	0.8960*	0.08%
Ru-En	0.3928	0.4208	2.80%	**0.4394**	4.66%	0.7864	0.7994	1.30%	**0.8000**	1.36%
En-De	0.3996	0.4245	2.49%	**0.4467**	4.71%	0.4005	0.4310	3.05%	**0.4433**	4.28%

Results obtained for **word-level (*F1-scores*)** and **sentence-level (*Pearson Correlation (r)*)** QE tasks. **STL**: results from the model trained on a single task. **LS-MTL** and **Nash-MTL**: results from the models trained using Linear Scalarization and Nash-MTL approaches, respectively. [* indicates the improvement is not significant with respect to the baseline (STL) score.]

Results: Multi-Pair Setting

LP	Word-Level ($F1$)					Sentence-Level (r)				
	STL	LS-MTL	+/- %	Nash-MTL	+/- %	STL	LS-MTL	+/- %	Nash-MTL	+/- %
En-Mr	0.4013	0.4349	3.36%	**0.4815**	8.02%	**0.6711**	0.6514*	-1.97%	0.6704*	-0.07%
Ne-En	0.4902	0.5406	5.04%	**0.5560**	6.58%	0.7892	**0.8012**	1.20%	0.8001	1.09%
Si-En	0.5629	0.6392	7.63%	**0.7003**	13.74%	0.6653	0.6837	1.84%	**0.6957**	3.04%
Et-En	0.4348	0.4998	6.50%	**0.5082**	7.34%	0.7945	**0.7970***	0.25%	0.7963*	0.18%
Ro-En	0.4472	0.4925	4.53%	**0.5285**	8.13%	**0.8917**	0.8883*	-0.34%	0.8895*	-0.22%
Ru-En	0.3965	**0.4241**	2.76%	0.4211	2.46%	0.7597	0.7751	1.54%	**0.7772**	1.75%
En-De	0.3972	0.4253	2.81%	**0.4499**	5.27%	**0.4373**	0.4308*	-0.65%	0.4298*	-0.75%

Results obtained for **word-level (*F1-scores*)** and **sentence-level (*Pearson Correlation (r)*)** QE tasks. **STL**: results from the model trained on a single task. **LS-MTL** and **Nash-MTL**: results from the models trained using Linear Scalarization and Nash-MTL approaches, respectively. [* indicates the improvement is not significant with respect to the baseline (STL) score.]

Results: Zero-Shot Setting

LP	Word-Level					Sentence-Level				
	STL	LS-MTL	+/- %	Nash-MTL	+/- %	STL	LS-MTL	+/- %	Nash-MTL	+/- %
En-Mr	**0.3800**	0.3692*	-1.08%	0.3833	0.33%	0.4552*	0.3869	-6.83%	**0.4674**	1.22%
Ne-En	0.4175	0.4472	2.97%	**0.4480**	3.05%	0.7548	**0.7601**	0.53%	0.7560	0.12%
Si-En	0.4239	0.4250*	0.11%	**0.4407**	1.68%	0.6416	0.6434*	0.18%	**0.6447***	0.31%
Et-En	0.4049	0.4206	1.57%	**0.4291**	2.42%	0.5192	0.5583	3.91%	**0.5598**	4.06%
Ro-En	0.4179	0.4349	1.70%	**0.4420**	2.41%	0.5962	0.6104	1.42%	**0.6300**	3.38%
Ru-En	0.3737	0.3761*	0.24%	**0.3834**	0.97%	0.5286	0.5605	3.19%	**0.5812**	5.26%
En-De	0.3750	0.3763*	0.13%	**0.3768***	0.18%	0.3217	0.3227*	0.10%	**0.3305**	0.88%

Results obtained for **word-level (*F1-scores*)** and **sentence-level (*Pearson Correlation (r)*)** QE tasks. **STL**: results from the model trained on a single task. **LS-MTL** and **Nash-MTL**: results from the models trained using Linear Scalarization and Nash-MTL approaches, respectively. [* indicates the improvement is not significant with respect to the baseline (STL) score.]

Results: Consistent Predictions

LP	Pearson Correlation (r)			Spearman Correlation (ρ)		
	STL	Nash-MTL	+/-	STL	Nash-MTL	+/-
En-Mr	-0.2309	-0.3645	**13.36%**	-0.1656	-0.2963	**13.07%**
Ne-En	-0.6263	-0.6604	3.41%	-0.6124	-0.6442	3.18%
Si-En	-0.5522	-0.5881	3.59%	-0.5380	-0.5510	1.30%
Et-En	-0.7202	-0.7539	3.37%	-0.7541	-0.768	1.39%
Ro-En	-0.7765	-0.7794	0.29%	-0.7380	-0.7534	1.54%
Ru-En	-0.6930	-0.7187	2.57%	-0.6364	-0.6805	4.41%
En-De	-0.4820	-0.5482	6.62%	-0.4524	-0.5099	5.75%

- Pearson (r) and Spearman (ρ) correlations between sentence-level and word-level QE predictions using STL and Nash-MTL QE models, in the single-pair setting.
- The **correlation** is computed between ***the z-standardized Direct Assessment (DA) scores*** and ***the bad tag counts*** normalized by sentence length.
- A stronger negative correlation denotes the predictions are more consistent.

Qualitative Analysis

Source	Target	STL	Nash-MTL	Label
[En] It is close to the holy site where the Buddha ages ago had turned wheel of Dharma and Buddhism was born.	**[Mr]** ज्या पवित्र स्थळावर शतकानुशतकांपूर्वी बुद्धांचा जन्म झाला होता, त्या जागेच्या जवळच हे मंदिर आहे.	0.25	-0.64	-0.64
[En] Representative species of the reserve include Bombax ceiba (Cotton tree), Sterculia villosa (Hairy Sterculia) and Cassia fistula (Golden shower tree).	**[Mr]** या संरक्षित क्षेत्राच्या प्रजातींमध्ये बोम्बॅक्स सिबा (कॉटन ट्री), स्टर्कुलिया विलोसा (हेरी स्टर्कुलिया) आणि कॅसिया फिस्टुला (गोल्डन शॉवर ट्री) यांचा समावेश आहे.	0.08	0.14	0.27
[Ro] Ulterior, SUA au primit mulți dintre elefanții africani captivi din Zimbabwe, unde erau supraabundenți.	**[En]** Later, the US received many of the captive African elephants from Zimbabwe, where they were overwhelming.	-0.02	0.81	0.95
[Ro] Aurul și argintul erau extrase din Munții Apuseni la Zlatna, Abrud, Roșia, Brad, Baia de Cris și Baia de Arieș, Baia Mare, Rodna.	**[En]** The gold and silver were extracted from the Apuseni Mountains in Zlatna, Abrud, Red, Brad, Baia de Cris and Baia de Arieș, Baia Mare, Rodna.	-0.37	0.67	0.83
[Si] පසුදා උදෑසන හෙලිකොප්ටර් යනා මගින් බලඝණ 2ක් ත්රිකුණාමලය ගුවන් කඳවුරට ගෙනයනලදී.	**[En]** Later in the morning, helicopter aircraft carried two powered triangular aircraft to the base.	0.43	-0.51	-1.03
[Si] අනෙකුත් ගොවීහු කෘෂිකර්මාන්තයේ විවිධ ක්රම අත්හදා බැලූ අය වූහ.	**[En]** Other farmers who experimented with various methods of agriculture.	-0.35	0.66	0.71

- The numbers in the **STL** and **Nash-MTL** columns are predictions (z-standardized DA scores) by the STL QE and Nash-MTL QE models, respectively. The **Label** column contains the ground truths.
- The MTL QE model predictions are more appropriate/justified than the STL QE model's predictions:
  - When a source sentence contains many named-entities.
  - When the translation is of high quality and only have minor mistakes.
  - When the source sentence (and therefore its translation) is complex.
- Both STL and MTL QE models are poor in predicting quality of sentences appropriately when a source sentence (and its translation) is in the passive voice.

Multi-Task Learning with APE and Quality Estimation

MTL-based Model for APE and QE

- Motivation:
  - To generate a high-quality output, an APE should know how much modification needs to be done and it should also be precise in identifying phrases in translation that need modifications.
  - Sentence-level QE predicts a DA score (real no.) that represents a quality of translation.
  - Word-level QE tags incorrect translation tokens with a ‘BAD’ tags.
  - We hypothesize that APE and QE are complementary to each other.
    - Sentence-level QE helps the APE system to understand how much correction is required, and the Word-level QE helps the APE in knowing where the corrections are required.
- Experiments: We jointly train an English-Marathi APE model on subsets of the following tasks:
  - APE, Word-level QE, Sentence-level QE (DA), Sentence-level QE (TER)
- Ablation study shows helpfulness of each QE task to APE

Results

- Test set: WMT22 APE Dev set

- LS-MTL: Model trained using Linear Scalarizatio n MTL approach
- Nash-MTL: Model trained using Nash-MTL approach

No.	Model	TER Scores
1	Do-nothing (Baseline)	22.93
2	LS-MTL (APE, Word-QE)	18.78
3	LS-MTL (APE, Sent-QE (DA))	19.52
4	LS-MTL (APE, Sent-QE (TER))	19.69
5	LS-MTL (APE, Word-QE, Sent-QE (DA))	18.54
6	LS-MTL (APE, Word-QE, Sent-QE (DA), Sent-QE (TER))	18.54
6	Nash-MTL (APE, Word-QE, Sent-QE (DA))	**18.30**

Summary and Conclusion

- We discussed our submission to the WMT22 English-Marathi APE Shared Task. Our approach shows the helpfulness of augmenting APE data with the phrase-level triplets. The results also show how we can use a sentence-level QE system to select the final output.

- Usefulness of QE for developing APE led us to developing better QE systems. We discussed how multi-task learning using Nash-MTL can improve performance of the QE model and also shows that jointly training a single model for different QE tasks results in consistent predictions.

- We extended this work and investigated whether MTL-based training helps APE models when trained with the QE tasks. We observe that Word-level QE and sentence-level QE (DA) are most helpful to APE.

References (1/4)

[1]: Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. Wav2vec 2.0: a framework for self-supervised learning of speech representations. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20)

[2]: OpenAI's Whisper: https://openai.com/blog/whisper/

[3]: [Honal and Schultz, 2003] Honal, M. and Schultz, T. (2003). Correction of disfluencies in spontaneous speech using a noisy-channel approach. In Interspeech. Citeseer.

References (2/4)

[8]: Anonymous, Jan. 2022. Seq-GAN-BERTSequence Generative Adversarial Learning for Lowresource Name Entity Recognition.

[9]: Shen et al, Feb. 2018. Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions

[10]: Saini, N. et al, Apr. 2021. Disfluency Correction using Unsupervised and Semi-supervised Learning. ACL 2021: Main Volume. Association for Computational Linguistics

[11]: Kundu, R. et al Oct. 2022. Zero-shot Disfluency Detection for Indian Languages. COLING 2022

References (3/4)

[1] Sourabh Deoghare and Pushpak Bhattacharyya. 2022. IIT Bombay's WMT22 Automatic Post-Editing Shared Task Submission. In Proceedings of the Seventh Conference on Machine Translation (WMT), pages 682–688, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

[2] Pushpak Bhattacharyya, Rajen Chatterjee, Markus Freitag, Diptesh Kanojia, Matteo Negri, and Marco Turchi. 2022. Findings of the WMT 2022 Shared Task on Automatic Post-Editing. In Proceedings of the Seventh Conference on Machine Translation (WMT), pages 109–117, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

[3] Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020. TransQuest: Translation Quality Estimation with Cross-lingual Transformers. In Proceedings of the 28th International Conference on Computational Linguistics, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

[4] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. 2022. Multi-Task Learning as a Bargaining Game. International Conference on Machine Learning (ICML), Baltimore, Maryland USA, July 17-23.

[5] Lucia Specia, Frédéric Blain, Marina Fomicheva, Chrysoula Zerva, Zhenhao Li, Vishrav Chaudhary, and André F. T. Martins. 2021. Findings of the WMT 2021 Shared Task on Quality Estimation. In Proceedings of the Sixth Conference on Machine Translation, pages 684–725, Online. Association for Computational Linguistics.

References (4/4)

[6] Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT Sentence Embedding. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

[7] 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.

[8] 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.

CS772: Deep Learning for Natural Language Processing (DL-NLP)

Fake-News & Half-Truth Detection

Presenter: Singamsetty Sandeep (M.Tech-II)
Computer Science and Engineering Department
IIT Bombay

Week 10 of 13th March, 2023

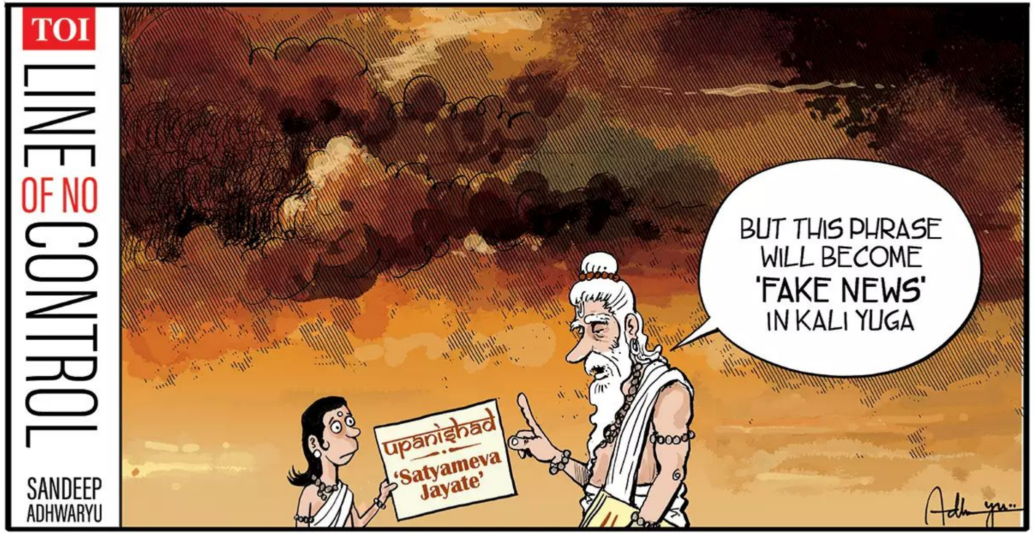
Topics of Discussion

1. **Topic:** Detecting and Debunking Fake News and Half-truth
   Presenter: Singamsetty Sandeep (M.Tech in CSE Dept.)
2. **Topic:** Query Intent Detection and Slot Filling
   Presenter: Apurva Kulkarni (IDDDP in CMINDS Dept.)
3. **Topic:** Speech Emotion Recognition
   Presenter: N V S Abhishek (M.Tech in CSE Dept.)

Introduction

Misinformation is spreading faster than ever and it is the easiest way to increase **viewership**, communicate with users, and **advertise** digitally.

Sources: social media, news channels, and digital platforms etc.



https://timesofindia.indiatimes.com/cartoons/humour/cartoons/line-of-no-control/age-of-fake-news/cartoonshow/69240778.cms

Fake News

Definition:
Fake news is false or misleading information that is presented as if it is true news.

https://www.indiatoday.in/india/story/top-ten-fake-news-that-we-almost-believed-in-2016-modi-best-pm-declared-unesco-359619-2016-12-26
https://www.facebook.com/kksfa/photos/a.151949284880275/392208987520969/?type=3

Half-truth

Definition:

A half-truth is a **deceptive statement** that contains some, but not all, elements of the truth. Half-truths are lies of omission. Even if a statement is technically true, when it leaves out crucial pieces of information, it can not be considered a truth.

Example: ***Electronic gadgets mandatory for e-census in 2023. (hidden: Govt. will provide the gadgets)***

- Will the government arrange those gadgets or common people should buy? (Deception)

Half-truth examples

1. ***I have never purchased a train ticket in my life to travel.***
   - The person might have not travelled in a train in his life. The above statement may be totally true, but it is misleading by hiding the truth.

2. ***People in Cuba are stinging themselves with blue Scorpions.***
   - People in cuba use an antidote made of poison from blue scorpion to boost immunity. The above statement is exaggerating that the people are stinging themselves with scorpions (literally).

3. ***Aswattama Hathaha! (Kunjaraha)***
   - Yudhisthir uses deception to confuse Dhronacharya that Ashwattama is dead, but slowly speaks that its an elephant named Ashwattama.

https://www.reuters.com/article/us-health-cancer-cuba-scorpion-idUSKBN1OD2GH

Problem Statement- Part 1

Given a claim and the corresponding evidence from a trustworthy source, predict the veracity of the claim and produce counters or supports for the predicted veracity label. The counters or supports produced for the claim are called explanations.

Input: A claim and corresponding evidence.

Output: A veracity label and an explanation (support or counter)

Veracity Labels: *true, half-true, false, barely-true, mostly-true, pants-on-fire.*

Note: *Supports are produced if the label is true,mostly-true and Counters are produced if the label is half-true, false, barely-true and pants-on-fire.*

Problem Statement- Part 2

Given a claim and the corresponding evidence from a trustworthy source, edit the claim if the claim is half-true or false or barely-true.

Input: A claim and corresponding evidence.

Output: An edited claim.

Task overview with an example

Claim: ***The Dolphins stadium renovation will create more than 4,000 new local jobs.***

Evidence: *The mailer distributed by Miami First said that the Dolphins stadium renovation will "create more than 4,000 new local jobs. "The Dolphins based the number on a 2010 study of a $225 million project that concluded 3,740 jobs in Miami-Dade and Broward. Instead, they tacked on an extra 260 jobs to the new $350 million project and say that is conservative. The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions. The Dolphins say that some jobs would continue, but they have provided no details as to how many of those 4,000 jobs would extend beyond the construction phase. To get those jobs, the team would receive $379 million from the state and county over about three decades, and eventually pay back about $159 million. As to whether the jobs will be local, the team has set a goal to hire the vast majority of the workers from Miami-Dade County but there is no financial penalty if they fail to do so.*

Label: ***Half-true***

Counter: ***The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions.***

Edited Claim: **The Dolphins stadium renovation will create temporary jobs.**

Motivation

There are many downsides to the **spread of fake news and half-truth** since they can **disrupt social and economic harmony**. The **black box model** employed for the task of fact verification should be **reliable** and **trustworthy**. To achieve this, besides predicting the veracity of a claim, it is important to provide counters or supports for the predicted label. Besides this editing the original claim if it is fake or half-true, will be helpful to **transform fake news into true news**. This would enable us to **counter the half-truth or fake news** and support the truth.

Literature Survey

1. **Guo et al. (2022)** presents an overview of the models and the datasets that exist in the domain of fact-checking which lists out all the challenges in this domain and also presents future directions.
2. **Kotonya and Toni (2020)** presents a few techniques used for explaining the verdicts in automated fact-checking.
3. **Alhindi et al. (2018)** introduced the LIAR-PLUS dataset. We are competing with this paper for the detection of veracity. The accuracy of our system has outscored the LIAR-PLUS dataset paper.
4. **Atanasova et al. (2020a)** generates justification for the claim and this textual summary which is generated is considered as the explanation for the veracity label predicted for the claim. This idea that a textual summary or a sentence can be used as an explanation was taken from this paper and used wisely in our research.
5. **Gardner et al. (2020)** show the effectiveness of contrast sets by creating them for various datasets. This idea has been used to study counterfactuals. Later, the idea of debunking fake news using counterfactuals came up. Thus, this paper was good at providing us with some ideas and understanding.
6. **Atanasova et al. (2020b)** discusses a technique to generate adversarial examples (using the extended version of (the HotFlip algorithm) for the target label for each claim in the FEVER dataset.
7. **Ross et al. (2021)** is a semantically controlled text generation system that uses SRL tags smartly and creates contrast sets for various downstream tasks without separately training a model for each task.

LIAR-PLUS Dataset

Paper: Where is Your Evidence: Improving Fact-checking by Justification Modeling (Alhindi et al., 2018)

LIAR-PLUS

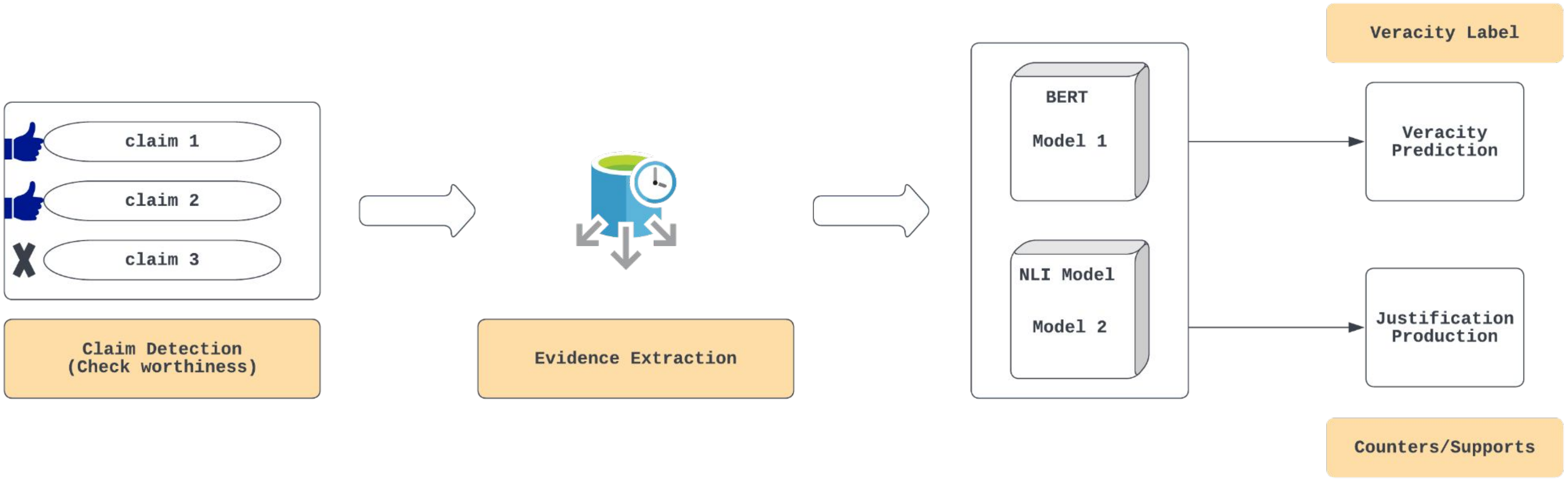
Extended version of LIAR dataset.

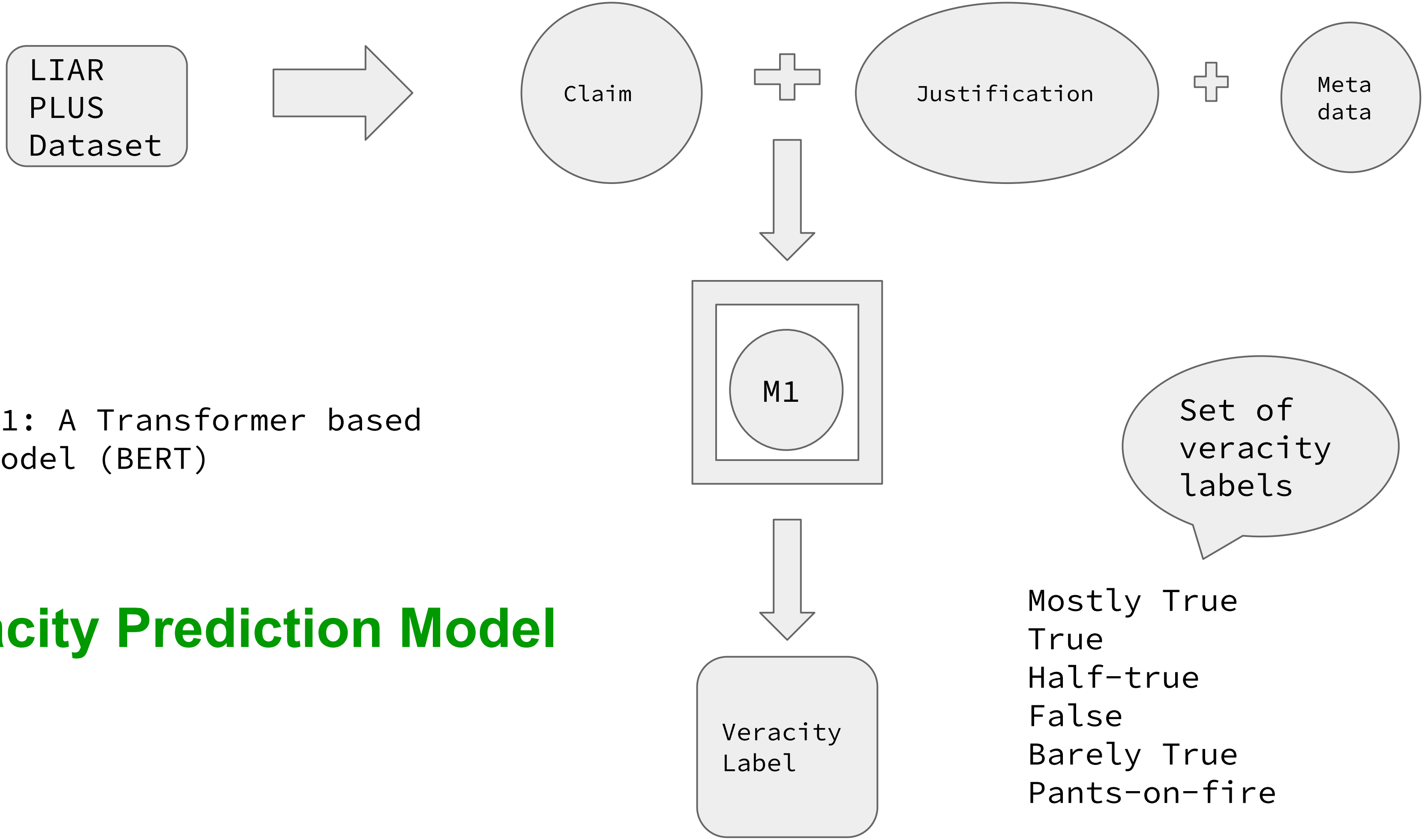
Reference: Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. 2018. **Where is your evidence: Improving fact checking by justification modeling.** *In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85–90, Brussels, Belgium. Association for Computational Linguistics.

- Column 1: the ID of the statement ([ID].json).
- Column 2: the label.
- Column 3: the statement.
- Column 4: the subject(s).
- Column 5: the speaker.
- Column 6: the speaker's job title.
- Column 7: the state info.
- Column 8: the party affiliation.
- Columns 9-13: the total credit history count, including the current statement.
  - 9: barely true counts.
  - 10: false counts.
  - 11: half true counts.
  - 12: mostly true counts.
  - 13: pants on fire counts.
- Column 14: the context (venue / location of the speech or statement).
- Column 15: the extracted justification

https://github.com/Tariq60/LIAR-PLUS

Fact Checking Pipeline



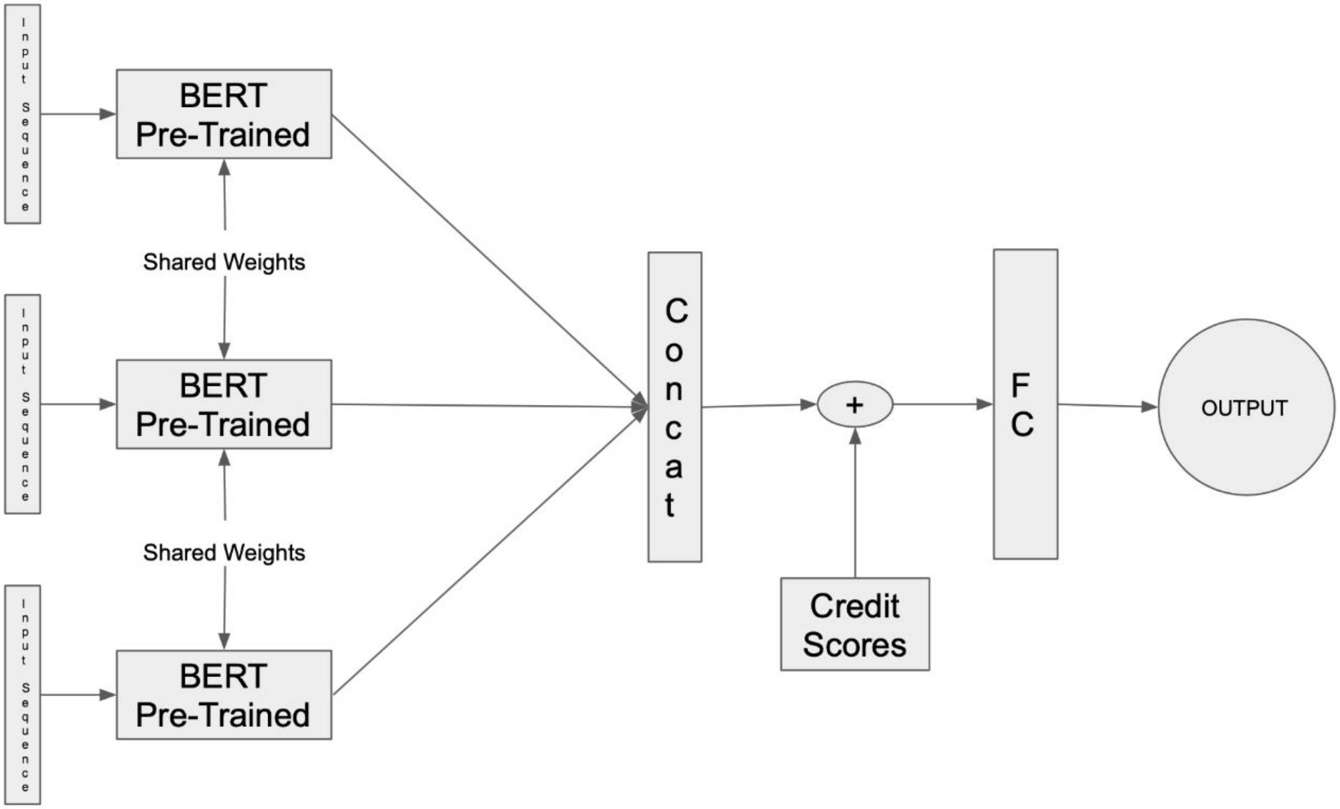


M1: A Transformer based
Model (BERT)

Veracity Prediction Model

Mostly True
True
Half-true
False
Barely True
Pants-on-fire

Architecture of Veracity Prediction Model



Task overview with an example

Claim: ***The Dolphins stadium renovation will create more than 4,000 new local jobs.***

Evidence: *The mailer distributed by Miami First said that the Dolphins stadium renovation will create more than 4,000 new local jobs. The Dolphins based the number on a 2010 study of a $225 million project that concluded 3,740 jobs in Miami-Dade and Broward. Instead, they tacked on an extra 260 jobs to the new $350 million project and say that is conservative. The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions. The Dolphins say that some jobs would continue, but they have provided no details as to how many of those 4,000 jobs would extend beyond the construction phase. To get those jobs, the team would receive $379 million from the state and county over about three decades, and eventually pay back about $159 million. As to whether the jobs will be local, the team has set a goal to hire the vast majority of the workers from Miami-Dade County but there is no financial penalty if they fail to do so.*

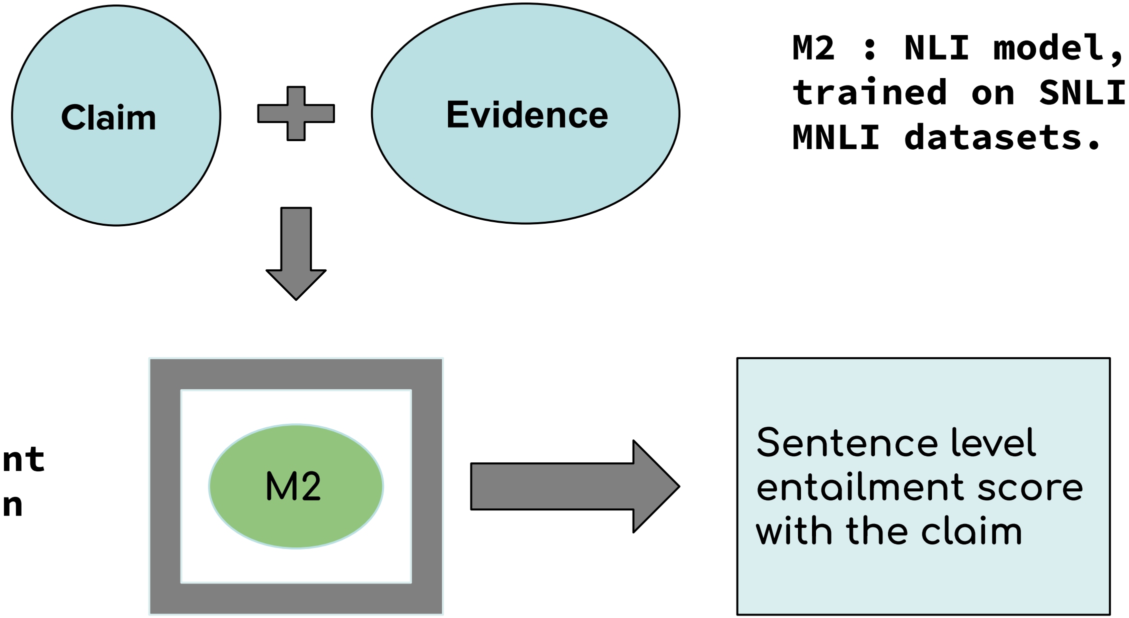
Label: ***Half-true***

Counter: ***The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions.***

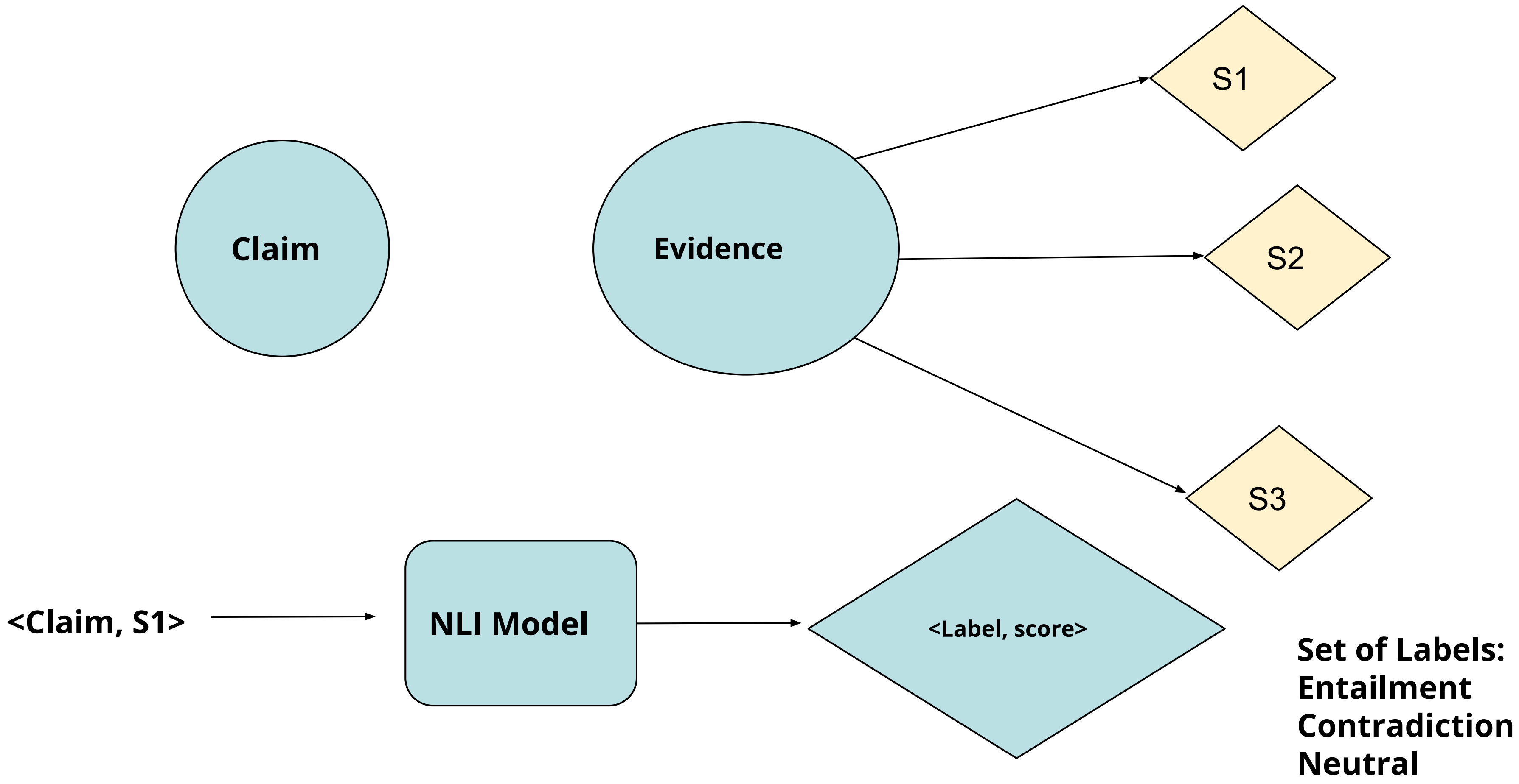
Edited Claim: **The Dolphins stadium renovation will create temporary jobs.**

Producing Explanations

M2 : NLI model, trained on SNLI and MNLI datasets.



M2: textual entailment detection model


Claim
Evidence
S1
S2
S3
<Claim, S1>
NLI Model
<Label, score>
Set of Labels:
Entailment
Contradiction
Neutral

Task overview with an example

Claim: ***The Dolphins stadium renovation will create more than 4,000 new local jobs.***

Evidence: *The mailer distributed by Miami First said that the Dolphins stadium renovation will create more than 4,000 new local jobs. The Dolphins based the number on a 2010 study of a $225 million project that concluded 3,740 jobs in Miami-Dade and Broward. Instead, they tacked on an extra 260 jobs to the new $350 million project and say that is conservative. The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions. The Dolphins say that some jobs would continue, but they have provided no details as to how many of those 4,000 jobs would extend beyond the construction phase. To get those jobs, the team would receive $379 million from the state and county over about three decades, and eventually pay back about $159 million. As to whether the jobs will be local, the team has set a goal to hire the vast majority of the workers from Miami-Dade County but there is no financial penalty if they fail to do so.*

Label: ***Half-true***

Counter: ***The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions.***

Edited Claim: **The Dolphins stadium renovation will create temporary jobs.**

Results

Method	2-Label accuracy	Six-Label accuracy
Dataset paper Alhindi et al. (2018)	70 %	37 %
BERT using Justifications	76 %	36.5 %
BERT using Supports and Counters	**81 %**	**39.5 %**

Table 6.1: Binary and Six-way classification accuracy of veracity prediction model

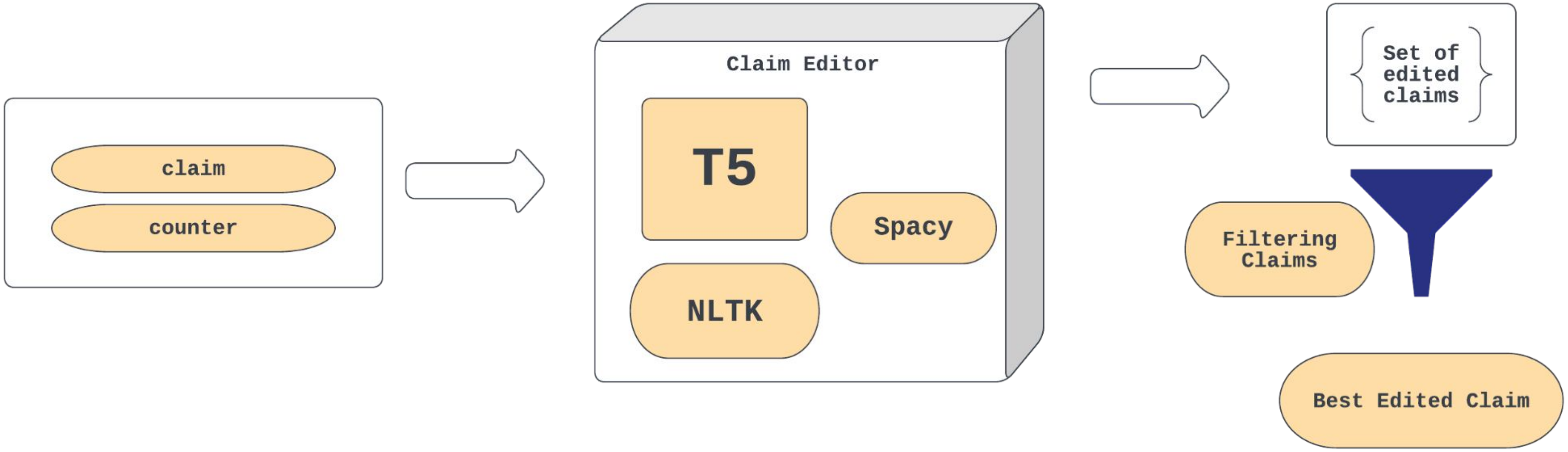
Qualitative Observations

- One of the most important observations that we have observed during the implementation of the veracity prediction model is the metadata is extremely useful to improve the accuracy of the model.

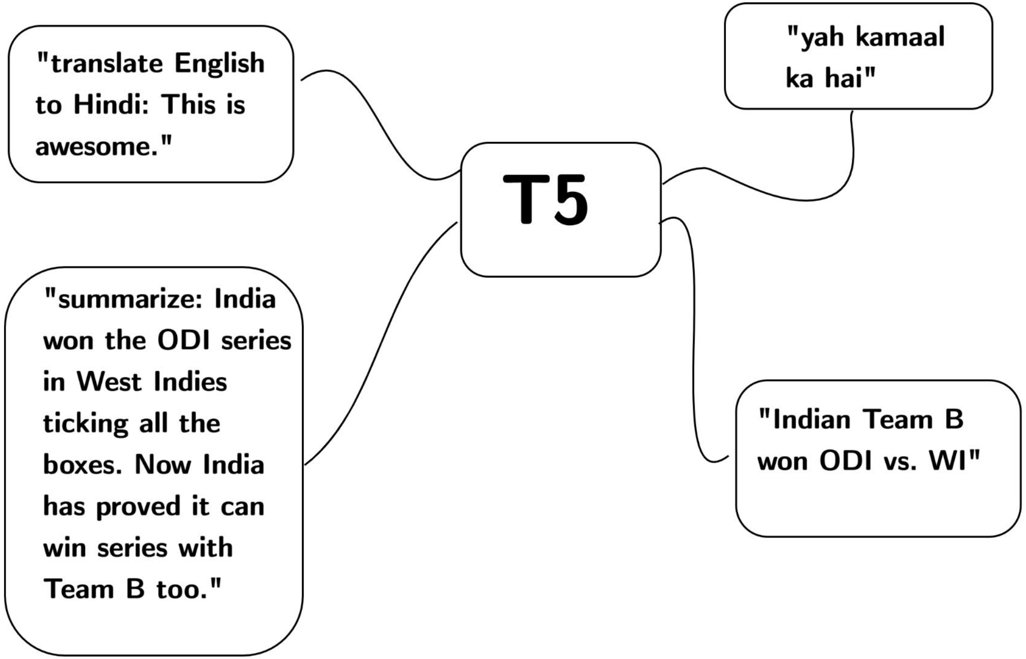
- We also observed that the length of the justifications is very long and in most cases, the justifications have useless additional information.

- We have extracted the relevant parts from the justifications in the LIAR-PLUS dataset using the NLI model. This has boosted the accuracy of the veracity prediction model.

Claim editing pipeline



T5 use cases


"translate English to Hindi: This is awesome."
"yah kamaal ka hai"
T5
"summarize: India won the ODI series in West Indies ticking all the boxes. Now India has proved it can win series with Team B too."
"Indian Team B won ODI vs. WI"

TAPACO Dataset - Paraphrase Dataset

TaPaCo dataset:

{
'paraphrase_set_id': '1483',
'sentence_id': '5778896',
'paraphrase': 'I ate the cheese.',
'lists': ['7546'],
'tags': [''],
'language': 'en'
}

A freely available paraphrase corpus for 73 languages extracted from the Tatoeba database. Tatoeba is a crowdsourcing project mainly geared towards language learners.

Scherrer, Yves. (2020). TaPaCo: A Corpus of Sentential Paraphrases for 73 Languages (1.0) [Data set]. Language Resources and Evaluation Conference (LREC), Marseille, France. Zenodo. https://doi.org/10.5281/zenodo.3707949

Original sentence: *Many people respect you. Do not disappoint them.*

Paraphrased sentence: *A lot of people look up to you. Do not let them down.*

Fine tuning T5

Input:

[[ARG0: A lot of people] [V: look] [ARG1: up to you] . Don’t let them down .]

Many people <extra_id_0> you. Don’t <extra_id_1> them.

Output:

Many people respect you. Do not disappoint them.

Masking Algorithm

The original **claim that needs an edit has to be masked**. We can't mask any token in the input since the claim has to be edited to make it true.

Hence for masked the right tokens, we used the idea of textual entailment and cosine similarity. We consider only the **parts of the claim that contradicts the evidence and has less similarity with evidence** to be replaced by a mask.

This aids the T5 model to fill the masks using the evidence and also guarantees that the right tokens are edited or replaced.

Task overview with an example

Claim: ***The Dolphins stadium renovation will create more than 4,000 new local jobs.***

Evidence: *The mailer distributed by Miami First said that the Dolphins stadium renovation will create more than 4,000 new local jobs. The Dolphins based the number on a 2010 study of a $225 million project that concluded 3,740 jobs in Miami-Dade and Broward. Instead, they tacked on an extra 260 jobs to the new $350 million project and say that is conservative. The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions. The Dolphins say that some jobs would continue, but they have provided no details as to how many of those 4,000 jobs would extend beyond the construction phase. To get those jobs, the team would receive $379 million from the state and county over about three decades, and eventually pay back about $159 million. As to whether the jobs will be local, the team has set a goal to hire the vast majority of the workers from Miami-Dade County but there is no financial penalty if they fail to do so.*

Label: ***Half-true***

Counter: ***The key omission here is that these are jobs associated with the 25-month stadium renovation project and include temporary positions.***

Edited Claim: **The Dolphins stadium renovation will create temporary jobs.**

Qualitative Observations

- Using the paraphrase dataset has aided the T5 model, in accurately learning the semantic and structural level properties of sentences which further aided its capacity to edit claims.

- Using the KL-divergence loss along with the original loss helped the T5 model preserve content and maintain fluency.

- Filtering the claims using a reward mechanism is extremely helpful and this has increased the overall accuracy of the claim editing pipeline.

Evaluation of edited claims

Model	BLEU Score	Content preservation	Perplexity
Tailor	0.82	0.75	**0.12**
GPT	0.58	0.49	1.12
ROBERTA	0.86	0.85	5.92
PEGASUS	0.18	0.42	1.20
Our Technique	**0.92**	**0.90**	2.27

Table 9.2: Evaluation of state-of-the-art models vs. our technique

Evaluation of edited claims on FAVIQ dataset

Model	BLEU Score	Content preservation	Perplexity
Tailor	0.84	0.78	**0.10**
GPT	0.56	0.52	1.18
ROBERTA	**0.92**	0.86	5.6
PEGASUS	0.16	0.44	1.4
Our Technique	0.90	**0.88**	2.4

Table 9.3: Evaluation of state-of-the-art models vs. our technique on the FAVIQ dataset

Evidence Extraction

Implemented a scraping bot to scrape news articles from from Google News.

Use case:

Claim: Virat Kohli to lead CSK for IPL 2023.

Evidence: Extracted from Google news

Link: https://www.dnaindia.com/cricket/

Former Indian skipper MS Dhoni is the second most followed cricketer on Instagram with 39.6 million followers, despite the fact that his last post on the photo-sharing app was on January 8, 2021. Dhoni is rarely active on any of the social media platforms and lives a simple life since his retirement from international cricket. He nonetheless continues to go strong for Chennai Super Kings (CSK) in IPL and will be seen leading them again in IPL 2023.

Google News Scraper

app home

half-truth

Your query: Virat Kohli to lead CSK for IPL 2023.

	media	title	subtitle
0	DNA India	Virat Kohli to MS Dhoni: Top 5 most followed cricketers on Instagram	Virat Koh
1	myKhel	IPL 2023 Auction Date, Retention Rules, Remaining Purse, Teams List - All You Need To Know	They eve
2	Sportstar	IPL Auction 2022 HIGHLIGHTS: 204 players sold for Rs 551.7 crore; Kishan, Chahar most expensive play	Lucknow
3	WION	`...from 1929 hrs`: When MS Dhoni sent fans into a meltdown with retirement from international crick	ALSO RE
4	The Cricket Lounge	There's A Clear Rift Between MS Dhoni And Ravindra Jadeja	Former I
5	The Cricket Lounge	MS Dhoni Has Finally Entered The World Of Movies	Dhoni, w
6	The Cricket Lounge	Pragyan Ojha Gave A Big Update About Virat Kohli's Future	Former I
7	Twelfth Man Times	IPL 2023: Ravindra Jadeja's childish antics on social media ...	The Indi
8	The Cricket Lounge	IPL 2022: Harbhajan Singh Names The Future Captain Of ...	He made
9	InsideSport.IN	IPL 2022 Auction: Top 5 highest paid Players in Retention	IPL 2022

Download data as Excel

Done!

Google News Scraper Demo

Experiment-1

Question: How many claims changed to true from half-true and false to true after claim editing?

Total claims: 2000

Half-true- 1000 and False- 1000

Results after claim editing:

Technique	Conversion to True statement
Edited Claims (Tailor)	1244 (62.2%)
GPT-2 PROMPT based	114 (5.7%)
ROBERTA (Text infilling)	75 (3.75%)
PEGASUS (Summary from evidence)	864 (43.2%)
T5 Claim editing model (Our model)	1694 (84.7%)

Experiment-2

Task: Use a baseline model to compare the accuracy of veracity prediction model.

Labels: True, False, half-true, mostly true, barely true, pants-on-fire

Logistic regression:

Input: Claim + Evidence (Counter or support)

Output: veracity label

Binary Classification: 76% (81% for BERT model)

- **Labels:** true,half-true,mostly-true- 1; barely-true,false,pants-on-fire- 0

Six-way Classification: 38% (39.5% for BERT model)

Half-truth binary classification: 72% (60% for BERT model)

- **Labels:** half-truth- 1 , all other labels- 0

Contributions

- We have generated supports and counters from evidence using textual entailment which boosted the accuracy of the veracity prediction model by 11% more than the highest reported accuracy in the LIAR-PLUS dataset paper.

- We devised a smart algorithm to mask the parts of a claim that needs to be edited which aided the T5 model to outperform cutting-edge systems such as GPT by 79%, Roberta by 81%, PEGASUS by 40%, and Tailor by 22% with a success rate of 85% for the task of claim editing.

- We implemented a real-time evidence extraction module using Google news scraper which is helpful for the verification of trending fake news.

Summary

We discussed fake news and half-truth with examples.

We have discussed the implementation of end-end fact checking pipeline.

We have discussed an interesting idea of debunking fake news using claim editing.

We have looked at the use case of Google News scraper to extract real time evidence.

We have discussed the experiments, results and evaluation of edited claims.

Conclusions

- The size and quality of the data play a prominent role in NLP to solve and tackle any problem. We have improved the quality of the LIAR-PLUS dataset by extracting only the **relevant pieces of information** from the evidence which **aided the improvement** in the accuracy of our veracity prediction model.

- Even though there are complex models and large models, such as GPT, a simple T5 model could outperform GPT for the task of claim editing. This proves time and again that for solving problems in NLP, models need to understand the **linguistic properties** better.

- A very powerful search engine like Google could not solve the problem with evidence extraction. We can't get the results of the relevant articles. Hence, search engines should be smarter in understanding figurative speech and complex language.

References

1. Alam, F., Shar, S., Dalvi, F., Sajjad, H., Nikolov, A., Mubarak, H., Martino, G. D. S., Abdelali, A., Durrani, N., Darwish, K., Al-Homaid, A., Zaghouani, W., Caselli, T., Danoe, G., Stolk, F., Bruntink, B., and Nakov, P. (2021). **Fighting the COVID-19 infodemic: Modeling the perspective of journalists, fact-checkers, social media platforms, policy makers, and the society.** *In Findings of EMNLP 2021.*
2. Alhindi, T., Petridis, $., and Muresan, $. (2018). **Where is your evidence: Improving fact-checking by justification modeling.** *In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 85-90, Brussels, Belgium. Association for Computational Linguistics.
3. Atanasova, P., Simonsen, J. G., Lioma, C., and Augenstein, I. (2020a). **Generating fact checking explanations**. *CoRR, abs/2004.05773.*
4. Atanasova, P., Wright, D., and Augenstein, I. (2020b). **Generating label cohesive and well formed adversarial claims**. *In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3168-3177, Online. Association for Computational Linguistics.
5. Barrén-Cedefio, A., Elsayed, T., Nakov, P., Martino, G. D. S., Hasanain, M., Suwaileh, R., Haouari, F., Babulkov, N., Hamdan, B., Nikolov, A., Shaar, S., and Ali, Z. S. (2020). **Overview of checkthat 2020: Automatic identification and verification of claims in social media.** *CoRR, abs/2007.07997.*
6. Chi, H. and Liao, B. (2022). **A quantitative argumentation-based automated explainable decision system for fake news detection on social media.** *Knowledge-Based Systems*, 242:108378.
7. Gardner, M., Artzi, Y., Basmova, V., Berant, J., Bogin, B., Chen, S., Dasigi, P., Dua, D.Elazar, Y., Gottumukkala, A., Gupta, N., Hajishirzi, H., Iharco, G., Khashabi, D., Lin, K., Liu, J., Liu, N. F., Muleaire, P., Ning, Q., Singh, S., Smith, N. A., Subramanian, S., Tsarfaty, R., Wallace, E., Zhang, A., and Zhou, B. (2020). **Evaluating NLP models via contrast sets.** *CoRR, abs/2004.02709*.

References

8. Jorrell, G., Kochkina, E., Liakata, M., Aker, A., Zubiaga, A., Bontcheva, K., and Dercaynski, L. (2019). **SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours.** *In Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845 854, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

9. Tuo, C., Cao, J., Zhang, X., Shu, K., and Yu, M. (2019). **Exploiting emotions for fake news detection on social media.** *CoRR, abs/1903.01728*.

10. Juo, Z., Schlichtkrull, M., and Vlachos, A. (2022). **A survey on automated fact-checking**. *Transactions of the Association for Computational Linguistics*, 10:178 206.

11. Gupta, P., Wu, C-S., Liu, W., and Xiong, C. (2021). **Dialfact: A benchmark for fact checking in dialogue.** *arXiv preprint* arXiv:2110.08222.

12. Kotonya, N. and Toni, F. (2020). **Explainable automated fact-checking: A survey**. *CoRR*, abs/2011.03870.

13. Li, Y., Zhang, J., and Yu, B. (2017). **An NLP analysis of exaggerated claims in science news.** *In Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 106-111, Copenhagen, Denmark. Association for Computational Linguistics.

14. Martinez-Rico, J. R., Martinez-Romo, J., and Araujo, L. (2021). **CheckThat! 2021: Check-worthiness estimation and fake news detection using transformer models**. In CLEP.

15. Park, J., Min, S., Kang, J., Zettlemoyer, L., and Hajishirzi, H. (2022). **FaVIQ: Fact verification from information seeking questions**. *In ACL*.

16. Ross, A., Wu, T., Peng, H., Peters, M. E., and Gardner, M. (2021). **Tailor: Generating and perturbing text with semantic controls.** *CoRR*, abs/2107.07150.

Thank you. 😊

CS772: Deep Learning for Natural Language Processing (DL-NLP)

Query Intent Detection and Slot Filling

Presenter: Apurva Kulkarni (IDDDP)
CMINDS
Department
IIT Bombay

Week 10 of 13th March, 2023

Query Intent Detection

Introduction

- Query Intent Detection is an important Information Extraction problem in NLP
- Classification problem - categorize an input user query among a set of specific intent classes
- Used to assist search engines by providing intent information of user queries to fetch appropriate results
- Important in tasks like virtual assistant services

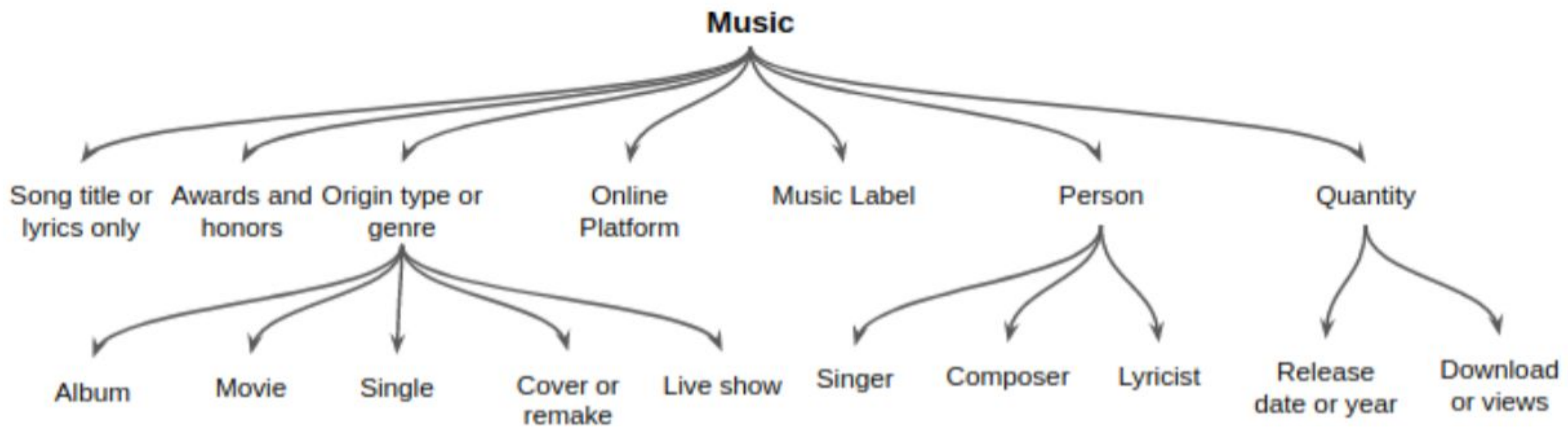
Problem Statement

- Classify user query into set of predefined intent classes
- Requires the creation intent class taxonomy for specific domains or tasks
- Taxonomy is created with the help of domain expertise while taking the downstream tasks into consideration
- Taxonomy can be multi level, with broad initial classes and finer subclasses

Example Taxonomy (1/2)

Level 1 Intent	Example Search Query
Movie	bahubali film dikhao
Music	purane songs ki list
TV or web-series	latest episode of tarak mehta ka ulta chasma
Social Media	baba ka dhaba viral
Celebrity	salman khan ka ghar
Books/Written Literature	mirza galib ke sher
Fashion	indore fashion week kab hota hai
Others	PS5 india mai kab launch hoga

Example Taxonomy (2/2)



Challenges

There are many challenges that can arise with this task in real world settings

- Multi-Domain - Queries from multiple domains with differing terminology and styles
- Multilingual - Support for multiple languages, code and script mixed queries
- Large number of classes, skew in data distributions and poorly represented classes
- Few shot or zero shot classification on unseen classes and domains

Approaches

- Initial approaches used text based features and classical ML models for query intent classification
- Deep learning revolutionised the field, end to end systems gave unmatched results
- The current state of the art involves use of deep learning architectures with pretrained language models
- These models are fine tuned on specific task data
- Transformer models like BERT give state of the art performances on most intent detection datasets

Multilingual Query Understanding For Entertainment Domain

Problem Statement

- Given a multilingual search query, identify the domain, intent, and entities in it, entities extracted transliterated to the native script.
- System developed for ‘Entertainment’ domain
- Language supported: Hindi, Marathi, Bengali, Tamil and Telugu
- Input can be code-mixed or script mixed with English
- System to be used to assist search engines for Indian languages

Challenges

- Language Challenges:
  - Transliteration - Delhi Bharat ki rajdhani hai
  - Code-mixing- दिल्ली इंडिया की कैपिटल है
  - Script-mixing- दिल्ली India की कैपिटल है
  - Structural orientation- Aamir khan movie songs
- Multilingual System: should support multiple languages
- No available Dataset and Taxonomy
- Test Data skew : large skew in the test datasets for domain classification to mimic the real-world data

Query Understanding Pipeline

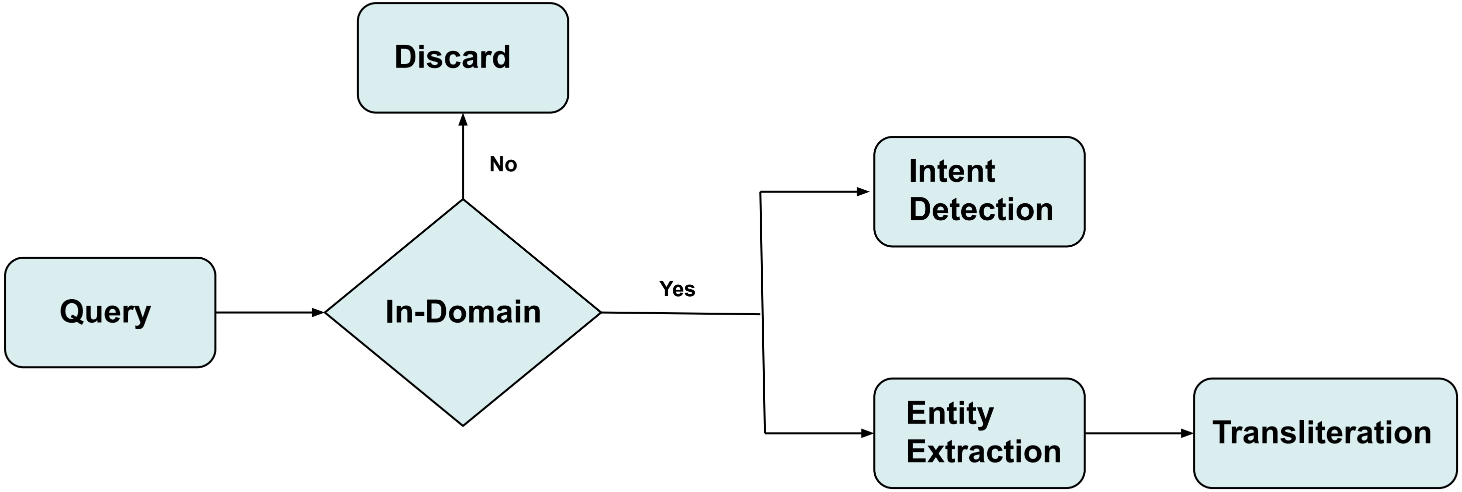
- A deep learning based query understanding pipeline -
  - Domain detection : binary classifier, which determines whether a query belongs to a particular domain or not
  - Intent detection : multiclass classifier to capture the intent of the query
  - Entity extraction : labels each word of the query to extract the entities from the query
  - Transliteration : transliterating all entities to the native script

Query Understanding Example

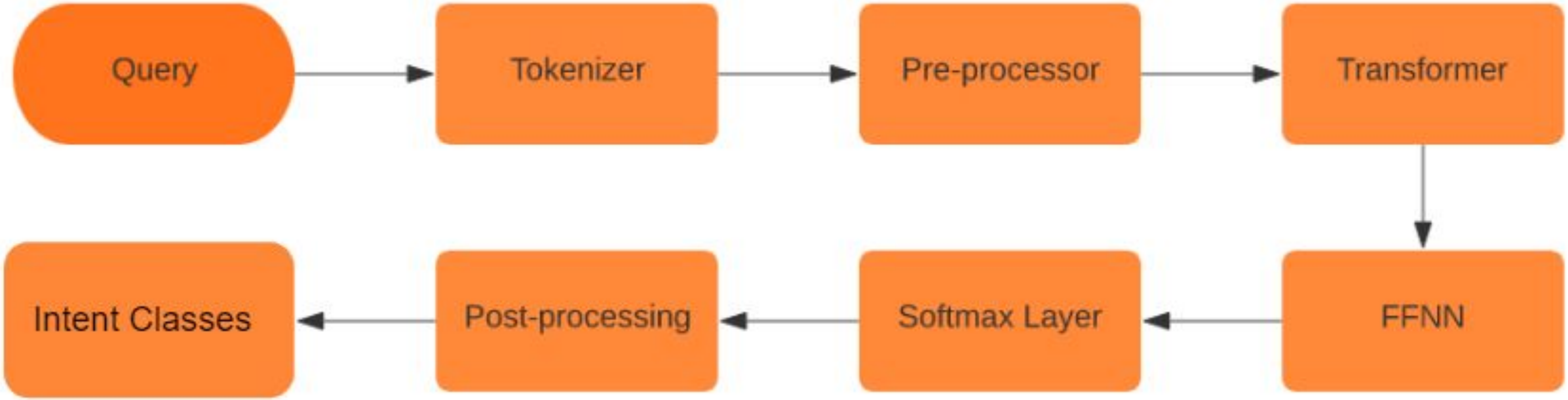
Example

- Query - Salman Khan ki nayi movie
  - Domain - Entertainment
  - Intent - movie-person-actor
  - Entities - Salman Khan
  - Transliteration - सलमान खान

Query Understanding Architecture



Intent Detection Model Architecture



Pretrained Models

- Language models like BERT are pretrained on large corpus (masked language modelling)
- These pretrained models are then fine tuned on specific downstream NLP tasks
- For our task we use the below mentioned pretrained BERT models, they are pretrained on all the languages that we are considering

Model:

- m-BERT or MuRIL , 12-layer, 768-hidden, 12-heads, 110M parameters

Vocabulary:

- Pre-train BERT has vocab size of ~110k for the 104 language
- Pre-trained MuRIL has vocab size of ~197k for 17 indian languages

Intent Taxonomy

- Taxonomy is a three level tree
- Level 1 intent is broad level or major intent consisting of 8 intent categories
- Levels 2 and 3 are further categorizations of their previous levels.
- This method creates 138 classes for intent
- Only 65 classes had sufficient representation and were considered
- Eg. query - endgame box office, intent - movie-quantity-earnings

Dataset

- Total number of queries annotated for intent is 47,475
- The taxonomy gives 65 classes for classification
- Dataset contains Hindi, Marathi, Bengali, Tamil and Telugu queries with code mixed and script mixed queries

Total Classes	Train	Val	Test
65	35,327	4,540	4,540

Multilingual Intent Detection Results

Test set	MuRIL (Original fine-tuned)		
Language	**L1**	**L1-L2**	**L1-L2-L3**
Hindi-English mixed	0.9191	0.7835	0.7589
Marathi	0.8334	0.6767	0.6462
Bengali	0.8054	0.6412	0.6096
Tamil	0.7778	0.5883	0.5515
Telugu	0.7671	0.5931	0.5586

Intent Detection and Slot Filling For Dialogue State Tracking

Dialogue State Tracking (DST)

- Information Extraction from user utterances in conversations with virtual assistants/chatbots that provide different services
- Dialogue State Frame - knowledge structure representing the kinds of intentions the system can extract from user sentences
- Frames - are a collection of slots that represent a type of information
- The system's goal is to fill the slots in the frame with the fillers the user intends, and then perform the relevant action for the user (answering a question, or booking a flight).

DST Slot Examples

- Air ticket booking service slots -

Slot	Type
ORIGIN CITY	city
DESTINATION CITY	city
DEPARTURE TIME	time
DEPARTURE DATE	date
ARRIVAL TIME	time
ARRIVAL DATE	date

DST System Architecture

A typical DST system has the following modules

- Domain Detection - determine broad category of user query, eg. airline booking, movie rentals, bank transaction service, hotel reservations, etc
- Intent Detection - determine general goal of user query, eg. find a flight, rent a chosen movie, set an alarm
- Slot filling - extract specific values for slots from user utterance

Slot Filling

- Slot FIlling - extract the particular slots and fillers that the user intends the system to understand from their utterance with respect to their intent.
- Eg. Show me morning flights from Boston to San Francisco on Tuesday
  - ORIGIN-CITY: Boston, ORIGIN-DATE: Tuesday, ORIGIN-TIME: morning, DEST-CITY: San Francisco

Datasets

- ATIS Dataset -
  - Queries on flight information from airline travel inquiry systems
  - 17 intent classes
  - 5000 train set instances, 800 dev and test set instances
- SNIPS Dataset -
  - 16000 queries from open domain
  - 7 intent classes
  - 13000 train set instances, 700 dev and test set instances

Multi-Domain Dataset

- MultiWOZ, Massive amazon
- Schema-Guided Dialogue Dataset -
  - 20k annotated multi-domain, task-oriented conversations between a human and a virtual assistant
  - Introduces complexity of different intent and slot classes in instances from different domains
  - Test set evaluation is zero shot - intent and slot filling classes are not seen during training

ATIS Intent Examples

Query	Intent Class
show me the fares from dallas to san francisco	atis-airfare
please give me flights available from baltimore to philadelphia	atis-flight
what ground transportation is there in atlanta	atis-ground service

ATIS Slot Filling Examples

Query	Slot filling tags
show me the fares from dallas to san francisco	O O O O O B-fromloc.cityname O B-toloc.cityname I-toloc.cityname
please give me flights available from baltimore to philadelphia	O O O O O O B-fromloc.cityname O B-toloc.cityname
what ground transportation is there in atlanta	O O O O O O B-cityname

SNIPS Intent Examples

Query	Intent Class
i want to listen to seventies music	PlayMusic
show me the picture creatures of light and darkness	SearchCreativeWork
i d like to go to the popular bistro in oh	BookRestaurant

SNIPS Slot Filling Examples

Query	Slot filling tags
i want to listen to seventies music	O O O O O B-year O
show me the picture creatures of light and darkness	O O O B-objecttype B-objectname I-objectname I-objectname I-objectname I-objectname
i'd like to go to the popular bistro in oh	O O O O O O O B-sort B-restauranttype O B-state

SGD Dataset example

Intent Detection Results

ATIS Intent Results

Model	Accuracy (%)	Macro F1 score (%)
Joint Bert (Chen et al., 2019)	97.9	-
Bi-model with decoder (Wang et al., 2018)	98.99	-
Our model	**98.62**	**93**

SNIPS Intent Results

Model	Accuracy (%)	Macro F1 score (%)
Joint Bert (Chen et al., 2019)	98.6	-
Our model	**97.87**	**97**

Slot Filling Results

ATIS Slot filling Results

Model	F1 score (%)
Joint Bert (Chen et al., 2019)	96.1
Bi-model with decoder (Wang et al., 2018)	96.89
Our model	**96.3**

SNIPS Slot Filling Results

Model	F1 score (%)
Bi-model with decoder (Wang et al., 2018)	93.8
Joint Bert (Chen et al., 2019)	97.0
Our model	**96.8**

Summary

- We introduced the problem of query intent detection and its applications
- We discussed the challenges and approaches for query intent detection
- We discussed a multilingual query understanding pipeline revolving around user intent detection for search engines for Indian languages
- We finally described the problems of intent detection and slot filling in dialogue state tracking

References(1/2)

1. J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In NAACL-HLT
2. Liat Ein Dor, Yosi Mass, Alon Halfon, Elad Venezian, Ilya Shnayderman, Ranit Aharonov, and Noam Slonim. 2018. Learning thematic similarity metric from article sections using triplet networks. pages 49–54
3. Teresa Gonalves and Paulo Quaresma. 2010. Multilingual text classification through combination of monolingual classifiers. CEUR Workshop Proceedings, 605.
4. Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages. CoRR, abs/2103.10730
5. Mukul Kumar, Youna Hu, Will Headden, Rahul Goutam, Heran Lin, and Bing Yin. 2019. Shareable representations for search query understanding
6. Barbara Plank. 2017. All-in-1: Short text classification with one model for all languages.
7. Pankaj Singh. 2021. Multilingual search query understanding
8. K Sreelakshmi, P Rafeeque, S Sreetha, and E Gayathri. 2018. Deep bi-directional lstm network for query intent detection. Procedia Computer Science, 143:939–946
9. Yu Wang, Yilin Shen, and Hongxia Jin. 2018. A bi-model based rnn semantic frame parsing model for intent detection and slot filling. pages 309–314.
10. Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling.

References(2/2)

11. Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling.

12. Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Mael Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces.

13. Yue Feng, Yang Wang, and Hang Li. 2021. A sequence-to-sequence approach to dialogue state tracking. pages 1714–1725.

14. Mukul Kumar, Youna Hu, Will Headden, Rahul Goutam, Heran Lin, and Bing Yin. 2019. Shareable representations for search query understanding.

Thank You

CS772: Deep Learning for Natural Language Processing (DL-NLP)

Speech Emotion Recognition

Presenter: N V S Abhishek (M.Tech-II)
Computer Science and Engineering Department
IIT Bombay

Week 10 of 13th March, 2023

Contents

Evolution of Intelligent Interactive Agents

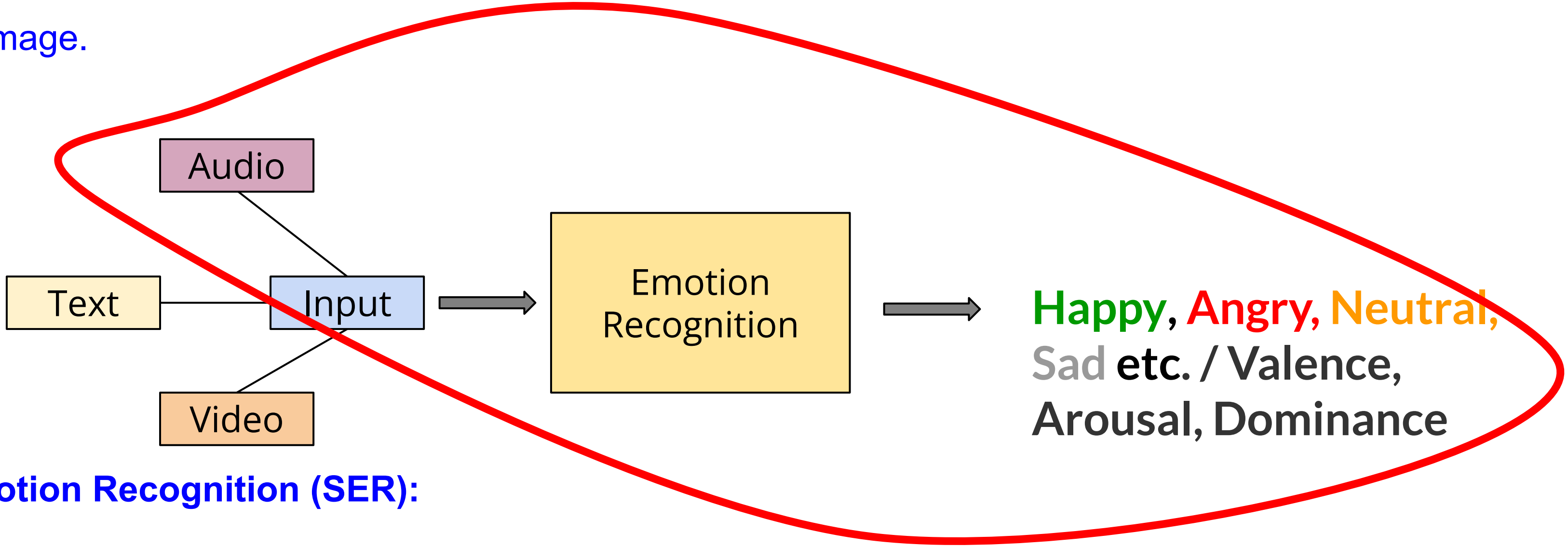
Conversational agents which can participate in a dialogue effectively have massive applications across multiple domains. *Mensio et al. (2018)* discussed three steps of evolution for conversational agents:

- Textual interaction
- **Vocal interaction**
- Embodied interaction

Mensio, Martino, Giuseppe Rizzo, and Maurizio Morisio. 2018. ***"The rise of emotion-aware conversational agents: threats in digital emotions."*** In Companion Proceedings of the The Web Conference 2018, pp. 1541-1544.

Emotion Recognition

Emotion Recognition is the task of predicting an emotion class (categorical model) like **happy**, **angry**, **sad** etc. or a real-valued metric like **valence**, **arousal** and **dominance** (dimensional model) for a piece of text, audio or image.



Speech Emotion Recognition (SER):

- Input: Audio Signal
- Output: Emotion Class

Emotion Recognition in Conversation (ERC)

- **Problem Statement:**
  - **Input**: Conversation with **N** utterances (Speech); **Output**: emotion label for each utterance
  - **$[(u_1, p_1), (u_2, p_2), \ldots, (u_N, p_N)]$** is a conversation with **N** utterances. Each utterance $\mathbf{u_i}$ is spoken by party $\mathbf{p_i}$
  - **$ui = [u_{i,1}, u_{i,2}, \ldots, u_{i,T}]$** consists of T words where $\mathbf{u_{i,j}}$ is the j^{th} word in the i^{th} utterance
  - The task of ERC is to predict the emotion label $\mathbf{e_i}$ of each utterance $\mathbf{u_i}$

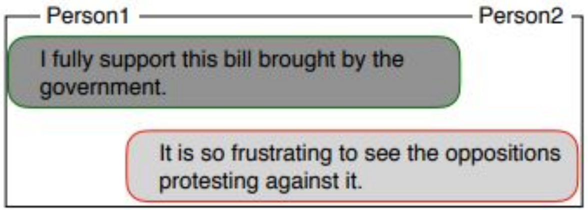
Controlling Variables in Conversation

- Conversations are governed by different factors or pragmatics, such as topic, interlocutors' personality, argumentation logic, viewpoint, intent, and so on.

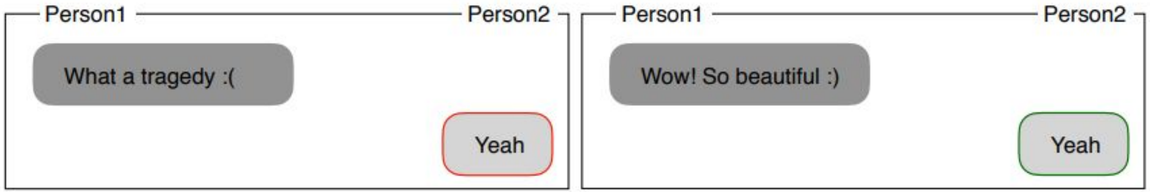
Poria, Soujanya, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019. "***Emotion recognition in conversation: Research challenges, datasets, and recent advances.***" IEEE Access 7: 100943-100953.

Challenges for ERC using Speech

- Lack of emotion-labeled speech data
- Noise and speech variations like accents make the task difficult
- Conversational context modeling
- Speaker specific modeling
- Listener specific modeling
- Presence of emotion shift
- Fine-grained emotion recognition
- Presence of sarcasm



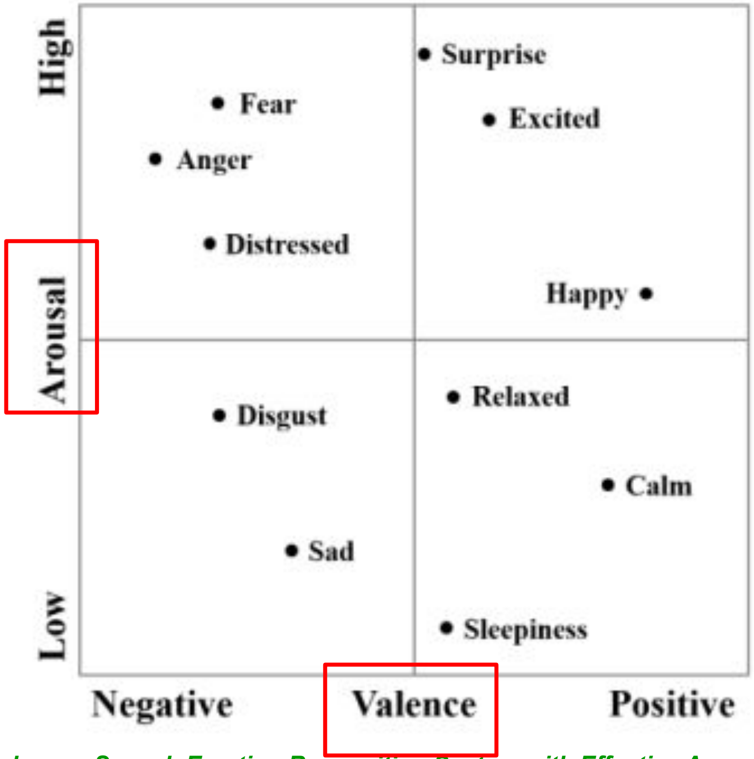
Fine-grained ERC



Importance of context in ERC

Poria, Soujanya, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019. "***Emotion recognition in conversation: Research challenges, datasets, and recent advances.***" IEEE Access 7: 100943-100953.

Two-Dimensional Emotion Model



Byun SW, Lee SP. 2021. “***A Study on a Speech Emotion Recognition System with Effective Acoustic Features Using Deep Learning Algorithms.”*** Applied Sciences, 11(4), 1890.

Plutchik’s Wheel of Emotions

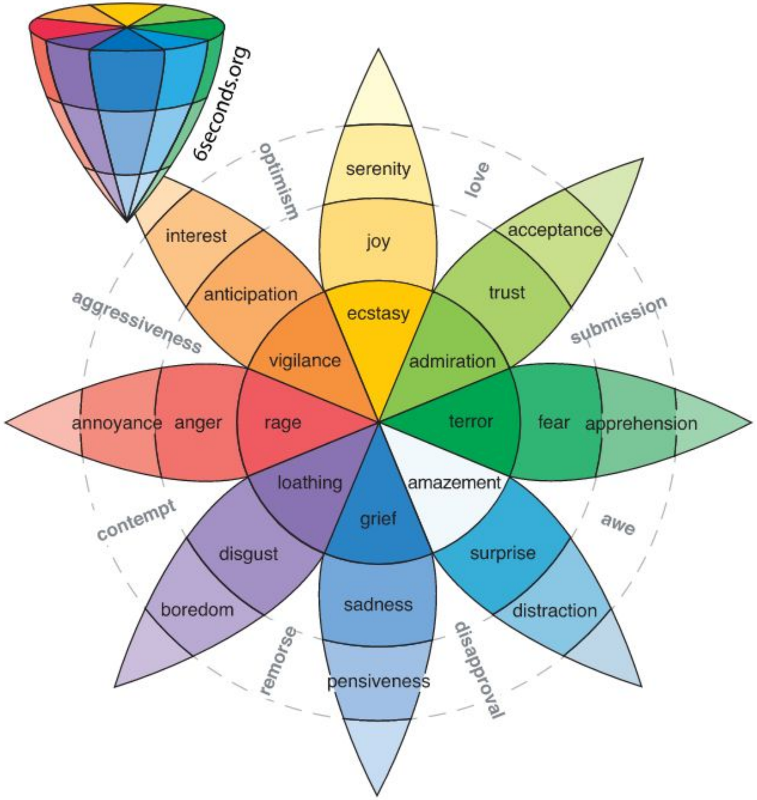
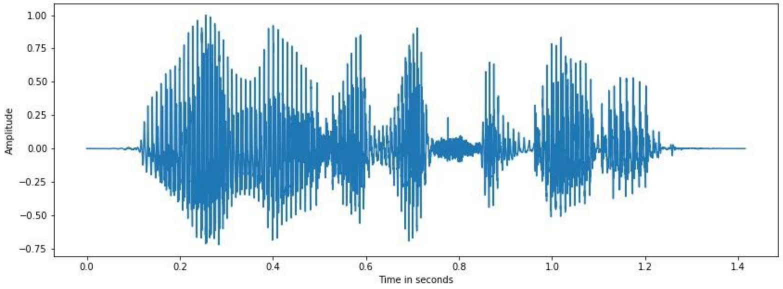


Image Credit:
https://www.6seconds.org/2022/03/13/plutchik-wheel-emotions/

Speech Features

Many low level acoustic features can be extracted from the raw audio signal.

- MFCC
- Energy
- Pitch
- Mel-Spectrograms



Picture of audio signal taken from https://towardsdatascience.com/understanding-audio-data-fourier-transform-fft-spectrogram-and-speech-recognition-a4072d228520

Automatic Speech Recognition

- Automatic Speech Recognition (ASR) is the task in which a speech utterance is converted to a sequence of tokens (words, subwords, characters).
  - Traditional Cascaded ASR Systems
    - Acoustic Model
    - Language Model
    - Pronunciation Model
    - Searching
  - End-to-End ASR Systems
    - Encoder-Decoder with Attention

Transformer-based Self-Supervised Architectures for Speech Processing Tasks

Learning Good Speech Representations: Wav2Vec 2.0 (2020)

- Self-supervised architecture which learns powerful speech representations from raw audio
- The feature encoder converts raw audio to latent representations
- The quantization module discretizes the feature encoder output for self-supervised training
- The **transformer** block gives contextualized speech representations
- In general there are two learning phases:
  - Self-supervised pre-training using large amount of unlabelled speech data
  - Supervised fine-tuning using smaller labelled speech data
- Authors used the following datasets:
  - Pre-Training: 53k hours of LibriVox (Crowd-sourced audio books collection) data
  - Fine-Tuning: Librispeech data

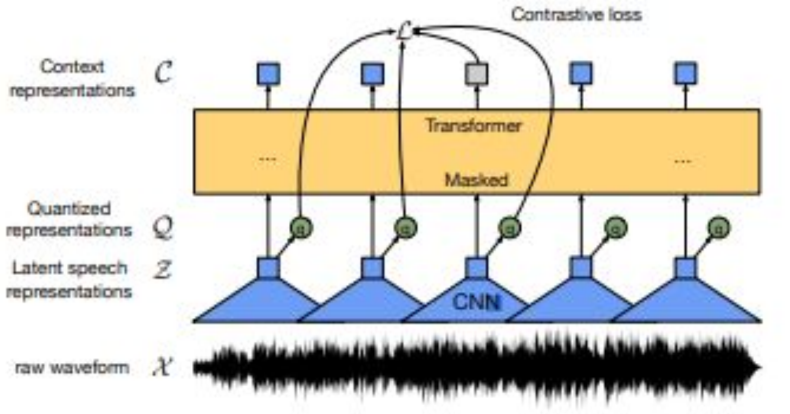


Image Credit: Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. ***wav2vec 2.0: A framework for self-supervised learning of speech representations***. Advances in Neural Information Processing Systems, 33:12449–12460.

Wav2Vec 2.0 Performance in ASR

- Results with different amount of fine-tuning data (for Librispeech clean/other test data):

Image Credit: Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in Neural Information Processing Systems, 33:12449–12460.

Question: Can wav2vec2 features do better than acoustic features for SER?

Methodology

- Prepare the audio dataset by resampling and extracting low level acoustic features from 5K audio files
- Extract wav2vec2 features for the 5K audio files
- Implement the architecture mentioned in *Pepino et al., (2021)*.
- Train the model in these settings:
  - Using only acoustic features (downstream-lla)
  - Using only wav2vec2 features (downstream-w2v2)
  - Using both low level acoustic and wav2vec2 features (downstream-lla_w2v2)
- Datasets: RAVDESS, TESS and CREMA-D

Model Architecture

- The figure shows the trainable parts in green.
- eGeMAPS is a Minimalistic acoustic feature set.
- (a) uses only w2v2 features

while (b) uses both w2v2 and eG2MAPS features.

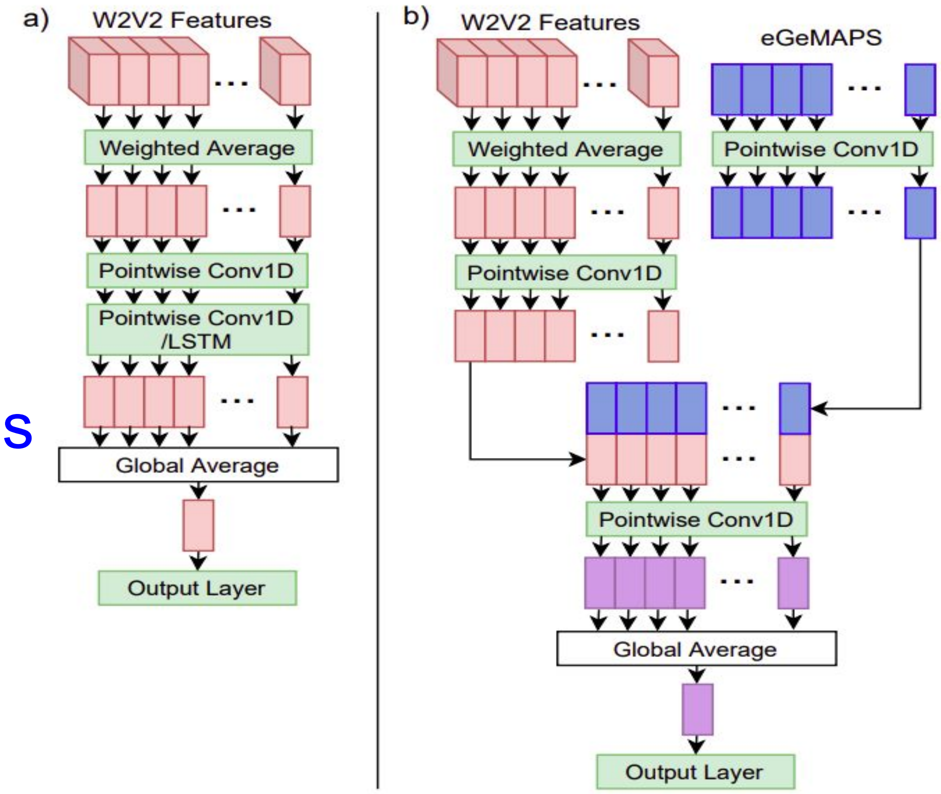


Image Credit: Pepino, L., Riera, P., Ferrer, L., 202
Emotion Recognition from Speech Using wav2vec
2.0 Embeddings. In: Proc. Interspeech 2021. pp.
3400–3404

Experimental Results

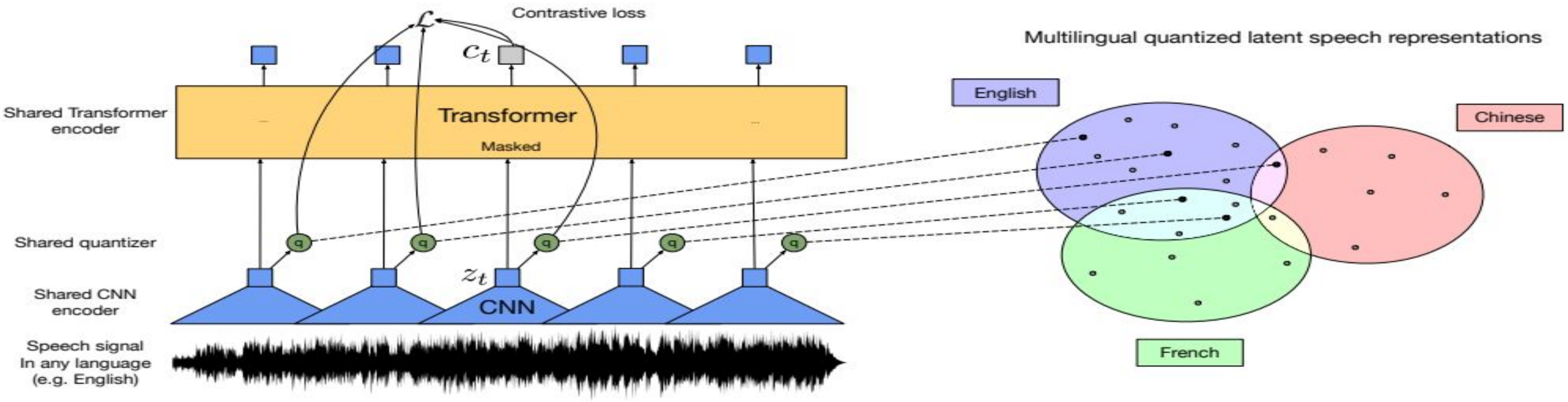
Model Name	Wt. Avg. F1 Score
wav2vec2-fine_tuned	0.79
downstream-lla	0.65
downstream-w2v2	0.75
downstream-lla_w2v2	0.77

Weighted avg. F1 score: Avg. of F1 scores of all the classes with each class's F1 score getting a weight equal to the number of samples of that class in the dataset.

<Code Explanation>

Learning Good Cross-Lingual Speech Representations: XLSR-Wav2Vec2

- Self-supervised architecture which learns powerful cross-lingual speech representations from raw audio
- A shared quantization module over feature encoder representations produces multilingual quantized speech units whose embeddings are then used as targets for a Transformer trained by contrastive learning.
- The model learns to share discrete tokens across languages, creating bridges across languages.
- Pre-Training: 56k hours of speech data from 53 languages (Combining **CommonVoice**, **BABEL** and **Multilingual Librispeech**)



Conneau, Alexis, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. 2020. ***"Unsupervised cross-lingual representation learning for speech recognition."*** arXiv preprint arXiv:2006.13979.

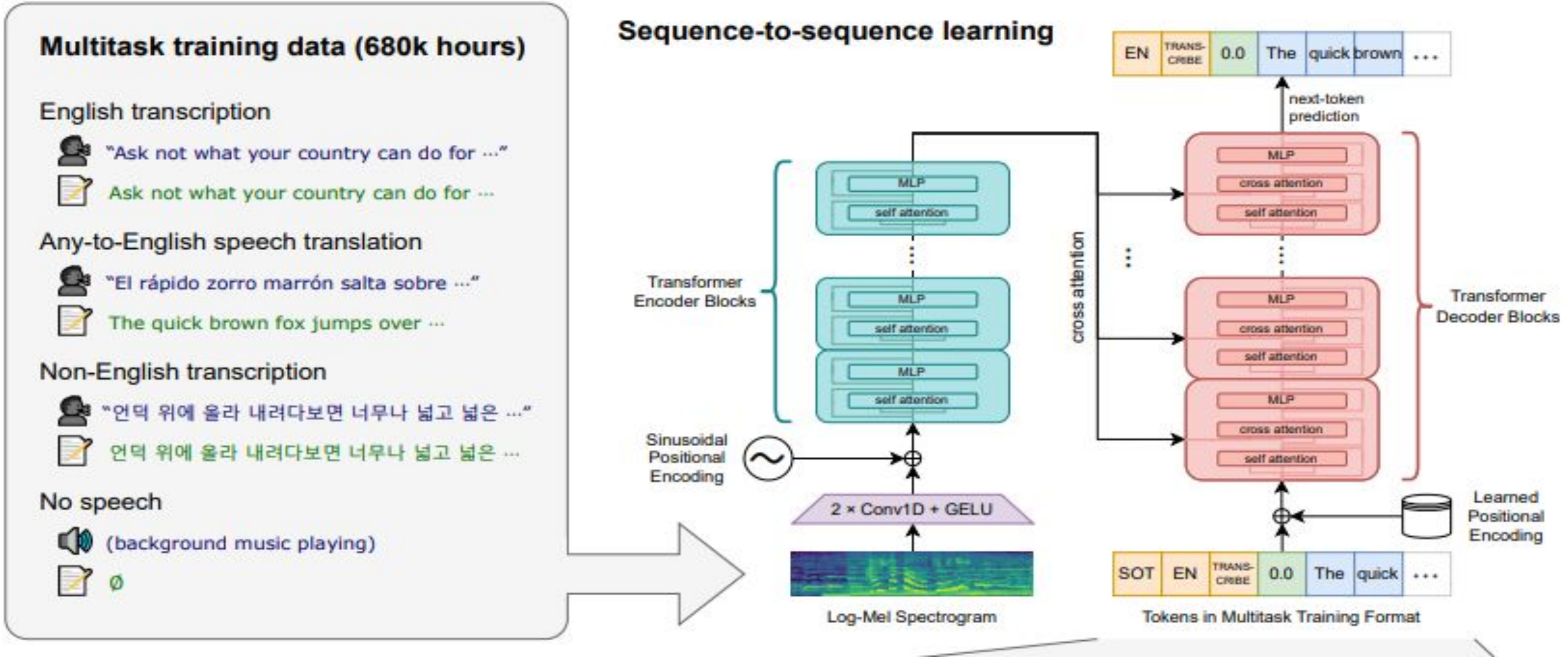
WHISPER by OpenAI (Sept. 2022)

- Weakly supervised technique to generate powerful speech representations
- 680K Hours of weakly supervised speech data (W2v2 uses 56K hours of unlabelled data) is used for training a simple Transformer architecture
- Multi-task approach:
  - Speech recognition
  - Language recognition
  - X speech -> English text
- WHISPER showed good generalizing capabilities close to human level performance for "out-of-distribution" data
- WHISPER is shown to be robust to variations like noise

Radford, Alec, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. ***Robust speech recognition via large-scale weak supervision. Technical report, OpenAI, 2022***. URL https://cdn. openai. com/papers/whisper. pdf,

WHISPER (Contd.)

- WHISPER Architecture



Radford, Alec, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. ***Robust speech recognition via large-scale weak supervision. Technical report, OpenAI, 2022***. URL https://cdn. openai. com/papers/whisper. pdf,

WHISPER (Contd.)

- WER of WHISPER when compared to Wav2Vec2
- Wav2Vec2 underperforms severely on “out-of-distribution” Data when compared to WHISPER

Dataset	wav2vec 2.0 Large 960h	Whisper Large
LibriSpeech test-clean	**2.7**	**2.7**
Artie	24.5	**6.7**
Fleurs (English)	14.6	**4.6**
Common Voice	29.9	**9.5**
Tedlium	10.5	**4.0**
CHiME6	65.8	**25.6**
WSJ	7.7	**3.1**
VoxPopuli (English)	17.9	**7.3**
AMI-IHM	37.0	**16.4**
CallHome	34.8	**15.8**
Switchboard	28.3	**13.1**
CORAAL	38.3	**19.4**
AMI-SDM1	67.6	**36.9**
LibriSpeech test-other	6.2	**5.6**
Average	29.5	**12.9**

Radford, Alec, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. ***Robust speech recognition via large-scale weak supervision. Technical report, OpenAI, 2022***. URL https://cdn. openai. com/papers/whisper. pdf,

SUPERB Benchmark

- SUPERB is a collection of benchmarking resources to evaluate the capability of a universal shared representation for speech processing.

* The four columns (1)~(4) correspond to the macs calculated with short, medium, long, longer bucket respectively
* Params = Parameter shared without fine-tuning

Method	Name	Description	URL	Params ↓	MACs ↓	(1) ↓	(2) ↓	(3) ↓	(4) ↓	Rank ↑	Score ↑	KS ↑	IC ↑	PR ↓	ASR ↓	ER ↑
WavLM Large	Microsoft	M-P + VQ ...		3.166e+8	4.326e+12	3....	6....	1....	2....	25.8	1145	97.86	99.31	3.06	3.44	70.62
WavLM Base+	Microsoft	M-P + VQ ...		9.470e+7	1.670e+12	1....	2....	4....	8....	24.05	1106	97.37	99	3.92	5.59	68.65
WavLM Base	Microsoft	M-P + VQ ...		9.470e+7	1.670e+12	1....	2....	4....	8....	20.95	1019	96.79	98.63	4.84	6.21	65.94
data2vec Large	Cl Tang	Masked G...		3.143e+8	4.306e+12	3....	6....	1....	2....	20.8	949	96.75	98.31	3.6	3.36	66.31
LightHuBERT Sta...	LightHuB...	Once-for-...		9.500e+7	-	-	-	-	-	20.1	959	96.82	98.5	4.15	5.71	66.25
HuBERT Large	paper	M-P + VQ		3.166e+8	4.324e+12	3....	6....	1....	2....	19.15	919	95.29	98.76	3.53	3.62	67.62
data2vec-aqc Base	Speech L...	Masked G...		9.384e+7	1.657e+12	1....	2....	4....	8....	19.05	935	96.36	98.92	4.11	5.39	67.59
CoBERT Base	ByteDanc...	Code Rep...		9.435e+7	1.660e+12	1....	2....	4....	8....	18	894	96.36	98.87	3.08	4.74	65.32
HuBERT Base	paper	M-P + VQ		9.470e+7	1.669e+12	1....	2....	4....	8....	17.75	941	96.3	98.34	5.41	6.42	64.92
wav2vec 2.0 Large	paper	M-C + VQ		3.174e+8	4.326e+12	3....	6....	1....	2....	17.7	914	96.66	95.28	4.75	3.75	65.64

SUPERB Leaderboard

Ongoing Work

Challenges for ERC

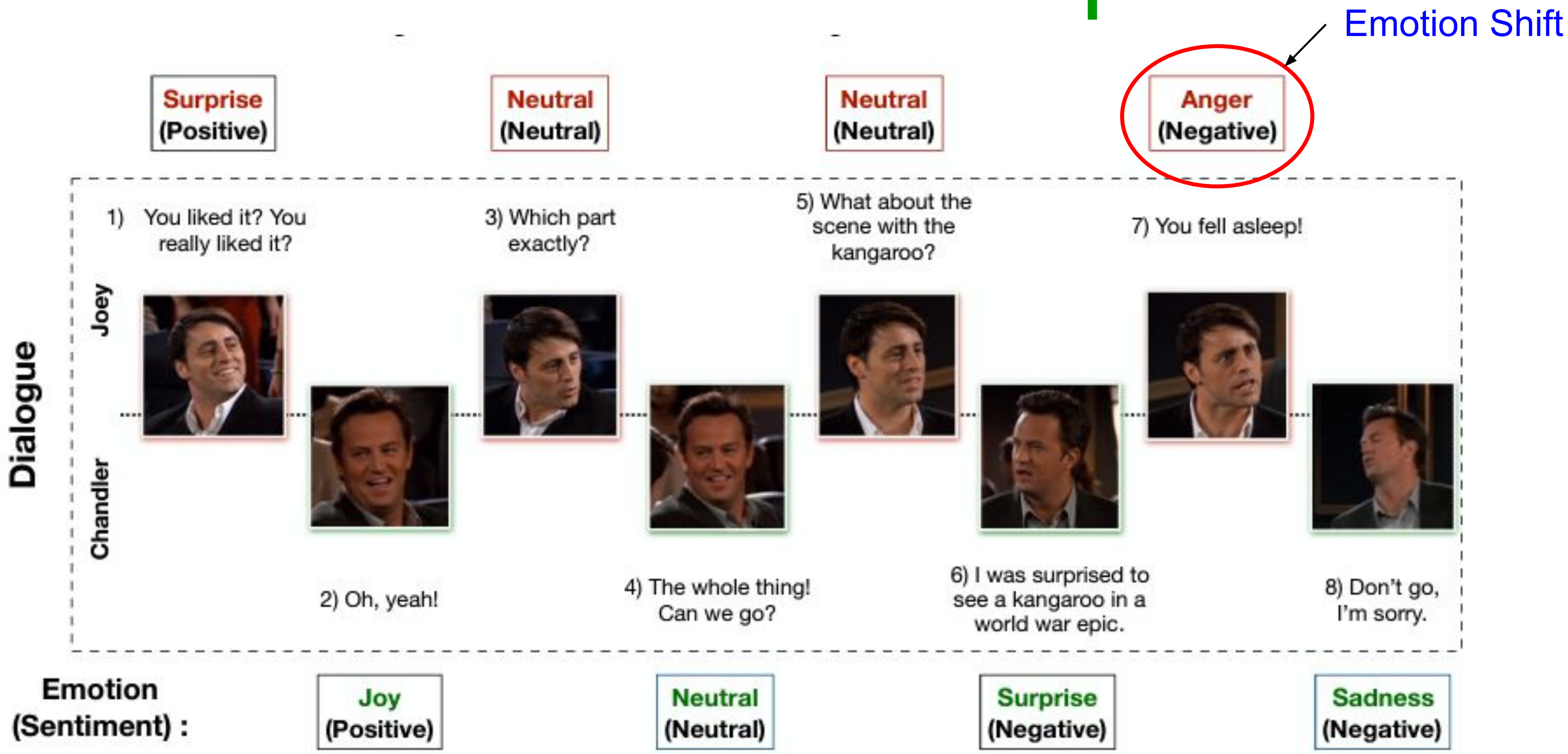
- Basis of emotion annotation
- Conversational context modeling
- Speaker specific modeling
- Listener specific modeling
- Presence of emotion shift
- Fine-grained emotion recognition
- Presence of sarcasm

	For Utterances Without Emotion Shift	For Utterances With Emotion Shift
Accuracy %age of Emotion Recognition	69.2%	47.5%

ERC performance of DialogueRNN

Poria, Soujanya, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019. "***Emotion recognition in conversation: Research challenges, datasets, and recent advances.***" IEEE Access 7: 100943-100953.

Emotion Shift Example

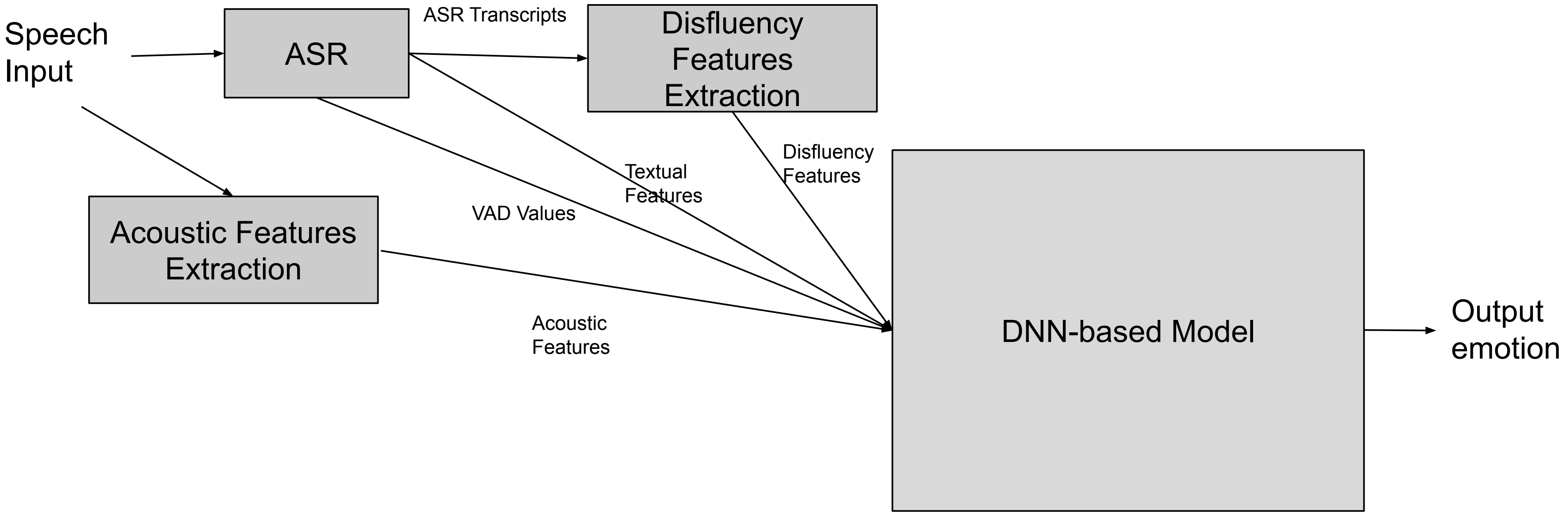


Poria, Soujanya, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019. "***Emotion recognition in conversation: Research challenges, datasets, and recent advances.***" IEEE Access 7: 100943-100953.

Speech ERC in a Natural Setting

- **Challenges**:
  - Variations in speech such as dialects, accents etc.
  - Noisy environments
  - Code-mixed and code-switched speech
- **Possible Approaches:**
  - Fine-tuning (Adequate Annotated Data)
  - Continual Pre-training (Adequate Unlabeled Data + Inadequate Annotated Data)

Overall Architecture for Proposed Approach



Summary & Conclusion

- Speech sentiment and emotion analysis is vital for coming up with good intelligent interaction systems.
- Pre-trained transformer-based models proved to be useful for the task of speech emotion recognition. Experimental results showed that using both wav2vec2 features and acoustic features are ideal for the task of speech emotion recognition.
- For multilingual SER XLSR-wav2vec2 can be utilised.
- Using WHISPER for SER can introduce robustness to SER models.
- Using disfluent features along with word-level VAD values of speech transcripts can enable us to detect emotion shift better.

References

- Pepino, L., Riera, P., Ferrer, L., 2021. ***Emotion Recognition from Speech Using wav2vec 2.0 Embeddings***. In: Proc. Interspeech 2021. pp. 3400–3404
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. ***wav2vec 2.0: A framework for self-supervised learning of speech representations***. Advances in Neural Information Processing Systems, 33:12449–12460.
- Shu wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y. Lin, Andy T. Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, Tzu-Hsien Huang, Wei-Cheng Tseng, Ko tik Lee, DaRong Liu, Zili Huang, Shuyan Dong, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung yi Lee. 2021. ***SUPERB: Speech Processing Universal PERformance Benchmark.*** In Proc. Interspeech 2021, pages 1194–1198.
- Chen, Sanyuan, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li et al. 2022. **"Wavlm: Large-scale self-supervised pre-training for full stack speech processing."** *IEEE Journal of Selected Topics in Signal Processing*.
- Conneau, Alexis, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. 2020. ***"Unsupervised cross-lingual representation learning for speech recognition."*** arXiv preprint arXiv:2006.13979.
- Gulati, A., Qin, J., Chiu, C.-C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., et al., 2020. “***Conformer: Convolution-augmented transformer for speech recognition”***. arXiv preprint arXiv:2005.08100

References

- Poria, Soujanya, Navonil Majumder, Rada Mihalcea, and Eduard Hovy. 2019. "***Emotion recognition in conversation: Research challenges, datasets, and recent advances.***" IEEE Access 7: 100943-100953.
- Zaijing Li, Fengxiao Tang, Ming Zhao, and Yusen Zhu. 2022. ***EmoCaps: Emotion Capsule based Model for Conversational Emotion Recognition***. In Findings of the Association for Computational Linguistics: ACL 2022, pages 1610–1618, Dublin, Ireland. Association for Computational Linguistics.
- Li, Yuanchao, Peter Bell, and Catherine Lai. 2022. "***Fusing ASR Outputs in Joint Training for Speech Emotion Recognition***." In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7362-7366. IEEE.
- Rovetta, Stefano, Zied Mnasri, Francesco Masulli, and Alberto Cabri. 2021. ***"Emotion Recognition from Speech: An Unsupervised Learning Approach."*** Int. J. Comput. Intell. Syst. 14, no. 1: 23-35.
- Sultana, Sadia, M. Shahidur Rahman, M. Reza Selim, and M. Zafar Iqbal. 2021. ***"SUST Bangla Emotional Speech Corpus (SUBESCO): An audio-only emotional speech corpus for Bangla."*** Plos one 16, no. 4: e0250173.
- Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., Narayanan, S. S., 2008a. “***Iemocap: Interactive emotional dyadic motion capture database”***. Language resources and evaluation 42 (4), 335–359

Thank You