Tackling Data Scarcity in Speech Translation Using Zero-Shot Multilingual Machine Translation Techniques

Authors: Tu Anh Dinh, Danni Liu, Jan Niehues

Department of Data Science and Knowledge Engineering
Maastricht University, The Netherlands

IEEE ICASSP 2022

Maastricht University

Overview

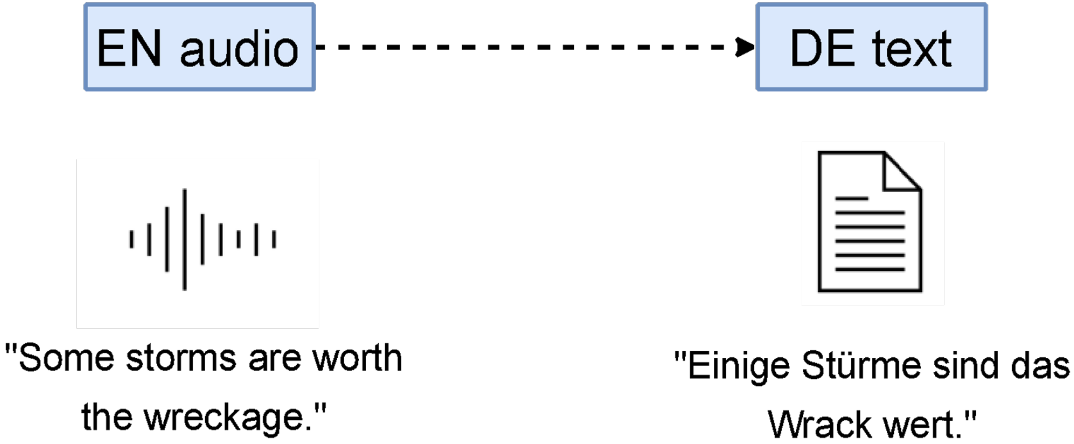
- Introduction
- Methods
- Experiments + Results
- Analysis
- Conclusions

Introduction

Motivation

Speech Translation (ST):

Translating speech in one language into text in another language



Introduction

Motivation

- Cascaded Speech Translation
  - Use 2 systems:
    - Automatic Speech Recognition (ASR)
    - Machine Translation (MT)

$$\boxed{\text{EN audio}} \xrightarrow{\text{ASR}} \boxed{\text{EN text}} \xrightarrow{\text{MT}} \boxed{\text{DE text}}$$

  - Problem: **error propagation**

Introduction

Motivation

Tackling error propagation:

- End-to-end Speech Translation
  - Use 1 system

    $$\boxed{\text{EN audio}} \xrightarrow{\text{End-to-end ST}} \boxed{\text{DE text}}$$

  - Problem: **lack of end-to-end ST data**

→ *Q: How do we tackle this ST-data-scarcity issue?*

Introduction

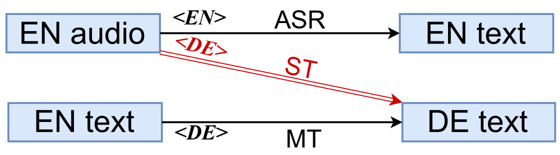
Proposed approach

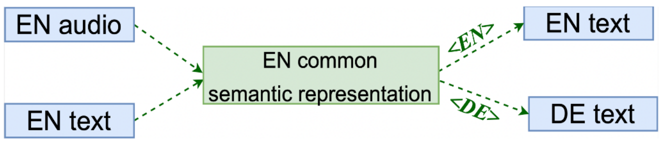
A: By leveraging ASR and MT data for training!

Contribution:

- End-to-end, multi-task model:
  - Trained on two tasks: ASR and MT
  - Fine-tuned with ST task
    → Few-shot models
  - Perform ST task during inference

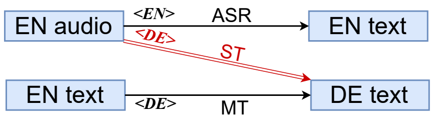
  Requirement: *Similar semantic representation across modalities* (*EN audio* and *EN text*)

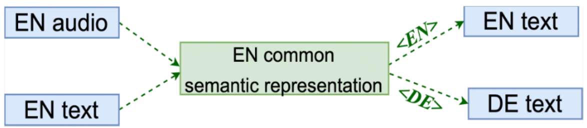




- Proposed methods:
  - Encouraging semantic similarity: auxiliary loss
  - Better control output language: data augmentation

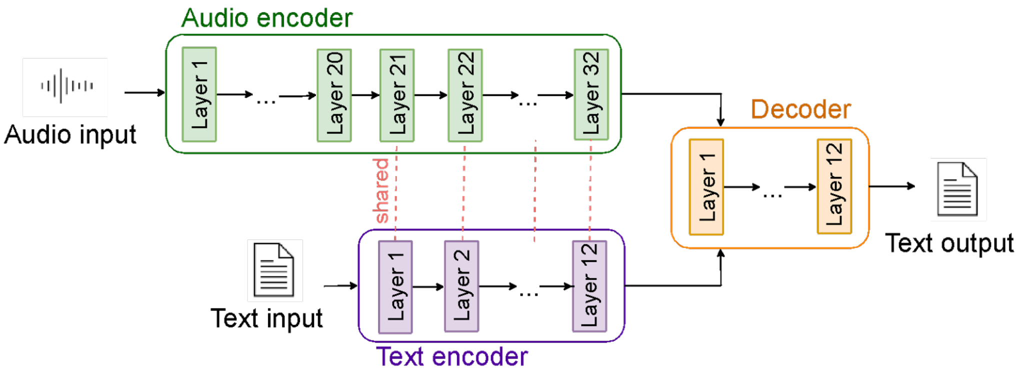
Methods





Base multi-task model

- Training data: ASR + MT
- Model architecture: Transformer
- 2 parallel encoders:
  - Text encoder + Audio encoder
  - Share parameters
    → Encourage similar semantic representation across modalities



Maastricht University

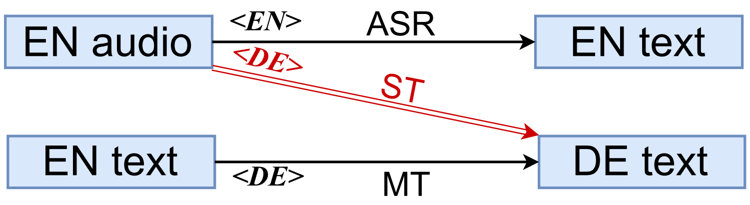
Methods

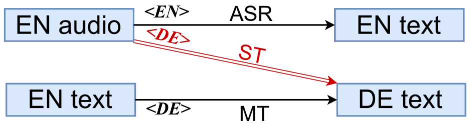
Base multi-task model

- Controlling output language:

  Add **target-language tokens** to:

  - the beginning of input sequences
  - every decoder input embedding





Methods

Cross-modality knowledge sharing: Auxiliary loss function

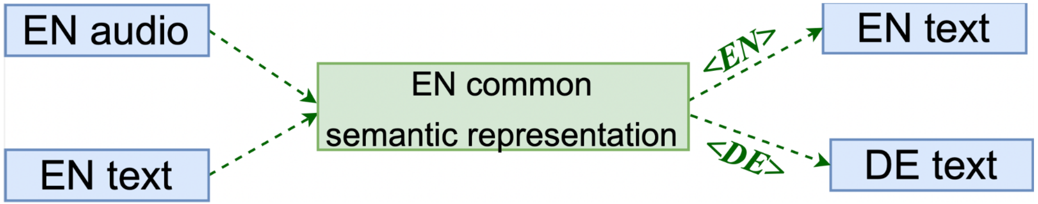
Auxiliary loss function

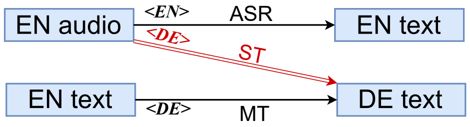
- Minimize text-audio encoder output difference between semantically similar sentences

  → **Modality-independent** representation
- Metrics for difference: squared error of mean-pool over time:

$$[mean\_pool(Encoder(X)) - mean\_pool(Encoder(Y))]^2$$

where X, Y are a pair of sentences with the same content, one in text and one in audio

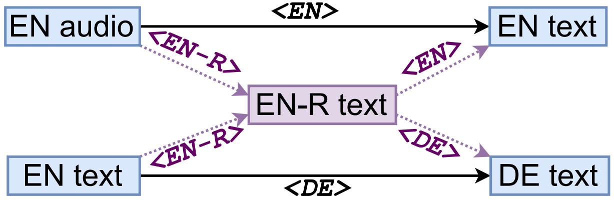




Methods

Better controlling output language: Data augmentation

- Problem:
  During training: Audio input → EN output
  Text input → DE output
  → Model decides on output language based on input modality, instead of the specified ***target-language token***
- Solution: data augmentation
  - Aim: having more than 1 target language output for each modality
    → Force the model to rely on ***target-language token***
  - Artificial language: character-wise-reversed English (EN-R)
    E.g. "Hello world!" → "Dlrow olleh!"
  - Require no additional real dataset

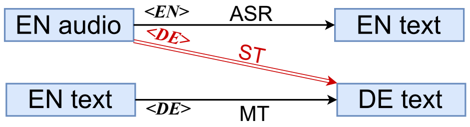


Experiment setups

- Data: CoVoST 2
  - A large-scale multilingual ST corpus
  - Focus of the paper: EN audio → DE text
    Data statistics:

	Training set	Validation set	Test set
Number of samples	289K	15K	15K

  - Models use all ASR and MT data for training;
    use 10% or 25% of ST data for fine-tuning
    → Few-shot models
- Reporting BLEU score on ST task (the higher the better)



Experiments + Results

Baseline models

	10% ST data for training/fine-tuning	25% ST data for training/fine-tuning
Direct end-to-end ST	0.5	0.8
Pre-trained with ASR	8.4	10.9
(Proposed model) Pre-trained with multi-task ASR and MT	**9.8**	**12.4**

- Direct end-to-end ST model not being able to perform ST task
- Model pretrained with ASR can perform ST task
- Proposed model gives the best performance
→ Strongest baseline

Experiments + Results

Proposed models

	10% ST data for fine-tuning	25% ST data for fine-tuning
Plain proposed model	9.8	12.4
Plain proposed model + auxiliary loss	10.6 **(+0.8)**	13.2 **(+0.8)**
Plain proposed model + augmented data	11.5 **(+1.7)**	13.5 **(+1.1)**
Plain proposed model + augmented data + auxiliary loss	11.5 **(+1.7)**	13.7 **(+1.3)**

- Auxiliary loss and data augmentation improves performance
- Most performance gain when used in combination
- More performance gain with less amount of ST data
  → Approaches particularly effective in low-resource scenarios.

Experiments + Results

Proposed models: comparison to full-data scenario

- Direct end-to-end model using 100% of ST data gives: 14.9 BLEU points
- Best proposed model using 25% of ST data gives: 13.7 BLEU points

→ Proposed model use significantly less ST data, yet only fail short by 1.2 BLEU points

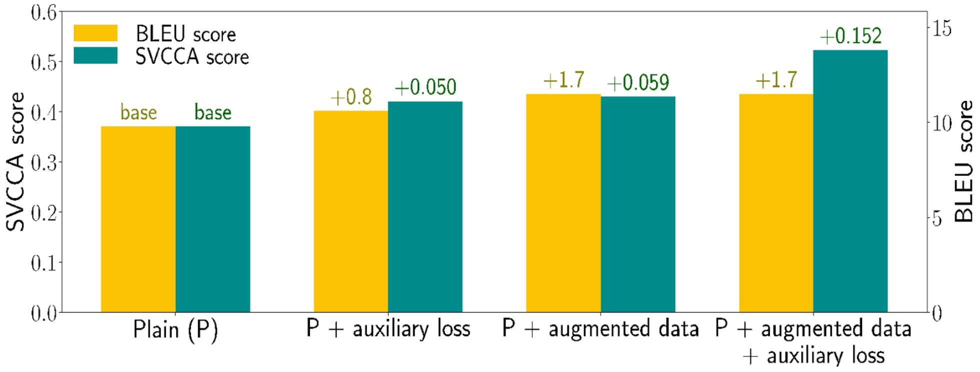
Analysis

Cross-modal similarity at sentence level and translation quality

- Singular Vector Canonical Correlation Analysis (SVCCA)
- *EN audio – EN text* meanpooled encoder output
- Higher SVCCA score
  ↔ More text-audio semantic similarity in **sentence level**

Observations:

- Proposed approaches increase text-audio similarity
- More text-audio similarity
  ↔ better ST performance



Maastricht University

Analysis

Cross-modal similarity at token level

- Classify encoder output tokens (text/audio)
- Better classification performance → lower text-audio similarity
- Outcome:
  - Models without auxiliary loss:
    Over 99.9% classification accuracy → two modalities very distinguishable
  - Models with auxiliary loss:
    Most tokens classified as “audio” → unable to distinguish two modalities

→ Auxiliary loss indeed improves **text-audio similarity** in **token level**

Conclusions

- Key requirement for leveraging ASR and MT data for ST task:
  Similar semantic representation across modalities

- ST performance improved:
  - Up to **+12.9** BLEU points vs. direct end-to-end ST models
  - Up to **+3.1** BLEU points vs. ST models fine-tuned from ASR models

- Proposed models successfully make use of ASR and MT training data for ST task

Thank you for your attention!