# Objective

To advance the field Speech Translation (ST) for Low Resource Languages (LRL).

# Immediate Objective

Reproduce the Baseline Model

# Domain

Low-Resource Languages

# Data Input

Audio Speech

Hopper Dir Location: /datasets/CS678

Data Set Name: CoVost 2

Data set URL: <https://commonvoice.mozilla.org/en/datasets>

Version: Common Voice Corpus 4.0

# Data Output

Text Translations

# Synopsis

A NATO Disaster Relief Emergency Response Team is deployed to the underdeveloped country where a Tsunami has just engulfed a local village. The worker must quick establish a relationship with local community leaders to coordinate and organize the deployment of relief supplies to the community. Unfortunately, the local community speaks a language that does not have a writing system nor an abundance of bilingual speakers. The relief worker was able to gather 20 hours of parallel speech-to-text data which was translated by one of bilingual speakers. He has one Titian GPX GPU. He wants to create a rough translation system where the speech of the local language can be translated into his local language writing system and vice-versa (optional).

# Approach

The approach for development of this system is composed of 3 phases.

|  |  |  |
| --- | --- | --- |
| Phase | Scheduled Time | Tasks |
| 1 | 3-10-2022 -> 10-14-2022 | 1. Data Acquisition 2. Data Preprocessing |
| 2 | 15-10-2022->28-10-2022 | 1. Develop Baseline Model |
| 3 | 29-10-2022->11-03-2022 | 1. Analyze the results |

Phase 1:

1. Identify the required data for the development of the model
2. Download the data for the model
3. Upload the data to Hopper GPU cluster
4. Develop code to do the preprocessing
   1. Develop the VAD Code
   2. Develop the MCFF Code
   3. Develop the Tokenization Code
   4. Produce summary statistic and analysis reports on the results from preprocessing

Phase 2:

1. Identify the high resource language pretrained model for the ASR Task
2. Identify the high resource language pretrained model for the MT task (if necessary)
3. Code the Transformer Architecture in pytorch
4. Code the Attention class
5. Code the embedding function (position+embedding+tk\_type\_ids)
6. Code the Add Norm function for the model
7. Code the feed forward layers function for the model
8. Code the CNN feature extractor for the model (if necessary)
9. Code the softmax layer for the model

Phase 3:

1. Analyze the output from the developed model and make sure they match the figures in Table2[[1]](#footnote-1).

# SDLC

The SDLC for each step will follow a modified version of the industry standards according to the water fall approach.

Figure 1. Industry Standard Waterfall

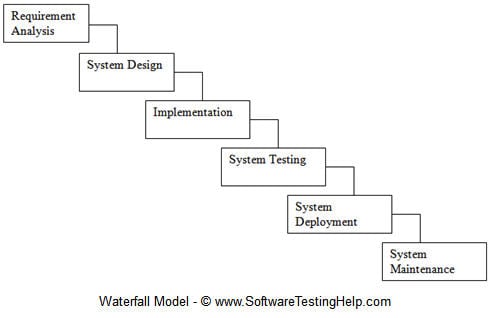


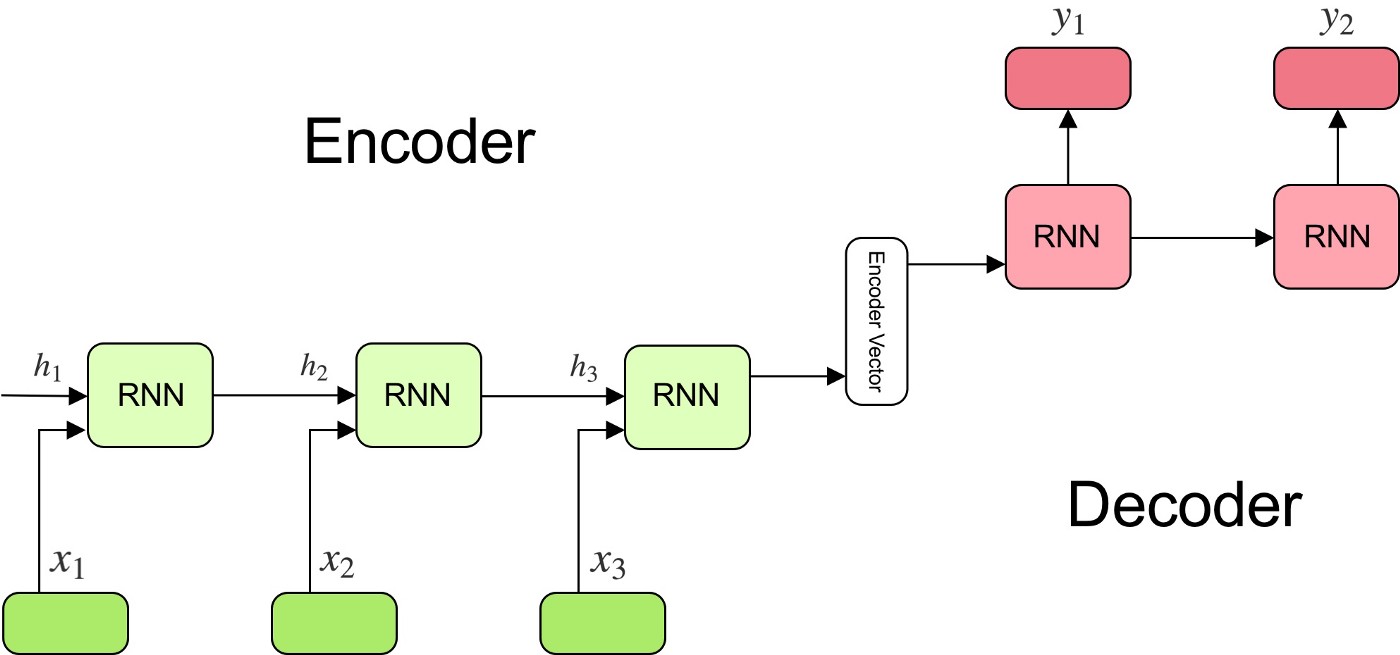
Figure 2. Modified Approach

Diagram

Description automatically generated

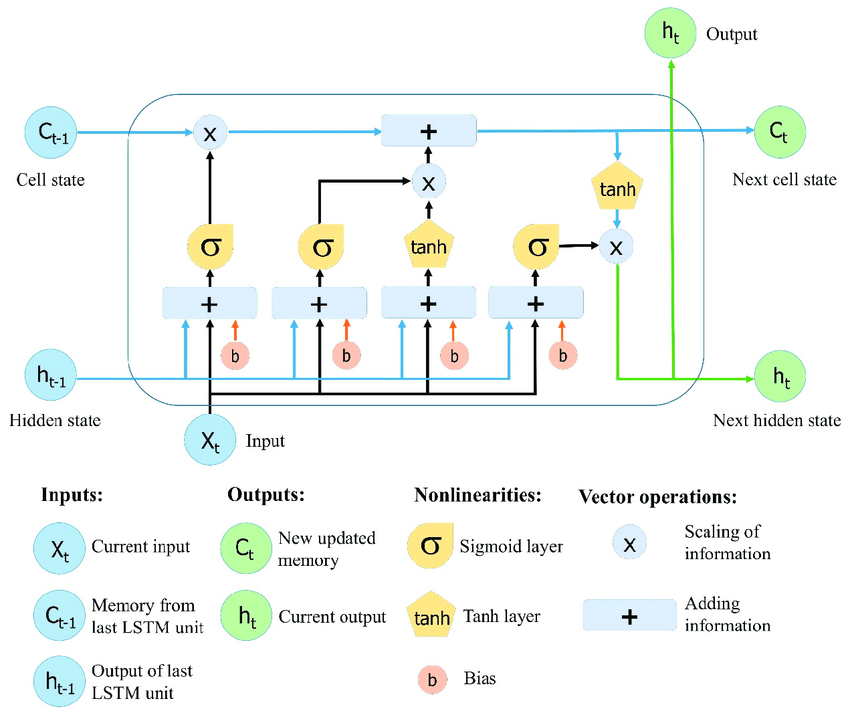
Diagram

Description automatically generated



Diagram

Description automatically generated



1. Changhan Wang, Anne Wu, and Juan Pino. 2020. Covost 2 and massively multilingual speech-to-text translation. arXiv preprint arXiv:2007.10310. [↑](#footnote-ref-1)