

MIDTERM PROJECT REPORT - FOUNDATIONS OF MACHINE LEARNING (CS725)

OCTOBER 31, 2023

Course Lecturer: Prof. Sunita Sarawagi

Yash Sunil Sadhwan	23M0818
Swapnil Bhattacharyya	23M0753
Ujjwal Sharma	23M0837
Saral Sureka	23M2113
Trivikram Vidya Umanath	23D1596
Sneha Oram	23M2159

Problem Statement

Our country is a diverse land of culture and language. In India there are 22 official languages being spoken. Translation in this case becomes a necessary task to inculcate inclusivity in people. The idea of machine translation was first conceptualized in 1954, and we have still not reached state-of-the-art in this regard. And, Language divergence is one of the main challenges that the field of machine translation struggles with.

The idea here is to develop machine translation to bridge the gap between two distinct families of language and enhance cross-lingual understanding.

Proposed Solution Approach

We try to develop Neural Machine Translation (NMT) for translating the English language to the Hindi language using diverse architecture combinations to optimize translation accuracy.

Training of NMT models is to be done using selected architectures (encoder-decoder with/without attention mechanism) and their performances are to be evaluated. Further, a comparative analysis is to be done for NMT-based translation outputs, assessing quality and efficiency. Initial models will be mostly based on simple encoder-decoder approach with LSTM/GRU and encoder-decoder model with attention mechanism.

Relevant Papers and Codebases

Following research papers and websites were taken as references

1. Kyunghyun Cho, Dzmitry Bahdanau, Fethi Bougarehttps: "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation".
2. Ilya Sutskever, Oriol Vinyals, Quoc V. Le. "Sequence to Sequence Learning with Neural Networks".
3. Dzmitry Bahdanau, KyungHyun Cho Yoshua Bengio. "NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSLATE".
4. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin. "Attention Is All You Need".
5. Translation with a Sequence to Sequence Network and Attention

Dataset

The dataset used for model building - IIT Bombay Hindi English Corpus

Dataset details -

<https://huggingface.co/datasets/cfilt/iitb-english-hindi>

It was decided at first to train the models with the entire dataset, but this was shooting up the training time. As the computation capacity was limited, we came up with trimming the data to contain 100,000 pairs. This was done by traversing the entire dataset, permuting randomly, selecting 100,000 examples, and filtering on length.

Dataset Statistics

- Trimmed the dataset to 97471 sentence pairs. Each pair is atmost of length 15.
- Total Words in the Vocabulary:
 - English : 52137
 - Hindi : 61973

```
[ ] input_lang, output_lang, pairs = prepareData('eng', 'hin', data_path)
    print(random.choice(pairs))

print("Total Sentences = ", len(pairs))
print(random.choice(pairs))

Reading lines... from eng-hin-train-100000.txt
Read 100000 sentence pairs
Trimmed to 97471 sentence pairs
Counting words...
Counted words:
eng 52137
hin 61973
['tea at the dhaba unning away from school !', 'सकल स भागना !']
Total Sentences = 97471
['rasgulla', 'रसगलल']
```

Figure 1: Dataset Statistics

This dataset was saved for further use. (see Fig. 1)

Implementation details

We have implemented the models using pytorch

Check out the code in GitHub Repository- <https://github.com/ujjwalsharmaIITB/FML-Project-NMT-En-Hi>

Vanilla Encoder-Decoder Model : A simple encoder-decoder architecture is a machine learning framework for sequence-to-sequence tasks. The encoder processes input data, creating a fixed-length context vector that summarizes the input. This context vector is passed to the decoder, which generates an output sequence, such as a translation. During training, the model learns by minimizing the difference between its generated output and the target sequence. This architecture is widely used in machine translation. (*Sequence to Sequence Learning with Neural Networks Ilya Sutskever, Oriol Vinyals, Quoc V. Le*)

Architecture

Encoder

Embedding layer input size

LSTM (also tried GRU) with hidden size of 128

Dropout layer

Encoder(

(embedding_layer): Embedding(52137, 128)

(rnn): GRU(128, 128, batch_first=True)

(dropout): Dropout(p=0.2, inplace=False)

)

Decoder

Embedding layer output size

LSTM (also tried GRU) with hidden size of 128

Dropout layer

Decoder(

(embedding_layer): Embedding(61973, 128)

(rnn): GRU(128, 128, batch_first=True)

(output layer): Linear(in_features=128, out_features=61973, bias=True)

)

Training

Trained of trimmed 100k dataset was done using Google Colab GPU.

Time Taken: 134 mins (see Fig. 2)

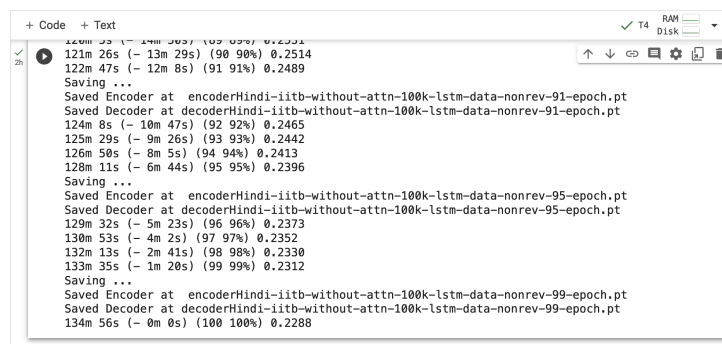


Figure 2: Training of the basic encoder-decoder model

Some inferences for this design

It was noticed that the model was not learning properly even though the loss was decreasing. It learned long translations and in some of them only initial words.

This behavior is expected and we suspect the following reasons

1. Not enough data was used.

2. Model was too simple, the hidden size was 128 (generally this is around 1024 or more). Also multiple encoders and decoders are used but we only had a single encoder and decoder.
3. No attention mechanism used, and so long range dependencies were forgotten.
4. Also because of teacher forcing the model learned initial words in a sentence but did not do well after that

Encoder-Decoder Model with Attention Encoder : Attention refers to a mechanism that allows a model to selectively focus on specific parts of the input data when making predictions or generating sequences. The attention mechanism assigns different weights to different elements of the input, emphasizing the more relevant elements while downplaying the less important ones. This dynamic weighting of input elements is particularly useful in tasks involving sequences, such as machine translation or text summarization, where certain parts of the input may be more relevant at different stages of processing. Attention helps models capture context and relationships within data, improving their accuracy and performance. In Attention mechanism the model automatically learns to (soft-)align and translate jointly.

(Neural Machine Translation by Jointly Learning to Align and Translate Dzmitry Bahdanau Jacobs University Bremen, Germany KyungHyun Cho Yoshua Bengio Universite de Montreal; <https://arxiv.org/abs/1409.0473>)

```
[ ] class BahadanuAttention(nn.Module):
    def __init__(self, hidden_size):
        super(BahadanuAttention, self).__init__()

        self.Wa = nn.Linear(hidden_size, hidden_size)
        self.Ua = nn.Linear(hidden_size, hidden_size)
        self.Va = nn.Linear(hidden_size, 1)

    def forward(self, decoder_hidden, encoder_hidden):
        align_scores = self.Va(torch.tanh(self.Wa(decoder_hidden) + self.Ua(encoder_hidden)))

        #             n*h*h + n*h*h = n*h*1
        #             # n*h n*1*h
        align_scores = align_scores.squeeze(2).unsqueeze(1)

        probabilisticWeights = F.softmax(align_scores, dim = -1) # n*1*h

        context_vector = torch.bmm(probabilisticWeights, encoder_hidden) # n*1*n = n * alphaij * hij

    return context_vector, probabilisticWeights
```

Figure 3: Bahadanau Attention

```
def forward_step(self, input_word, decoder_hidden, encoder_outputs):
    embedded_input = self.dropout(self.embedding(input_word))

    # hidden state is also called query
    hidden_state_as_query = decoder_hidden.permute(1,0,2)

    context_vector, attention_weights = self.simpleAttention(hidden_state_as_query, encoder_outputs)

    input_rnn = torch.cat((embedded_input, context_vector), dim=2)

    output, hidden = self.rnn(input_rnn, decoder_hidden)

    output = self.output(output)

    return output, hidden, attention_weights
```

Figure 4: Use of Attention

Encoder

Embedding layer input size

GRU with hidden size of 128

Dropout layer

```
Encoder(
  (embedding_layer): Embedding(52137, 128)
  (rnn): GRU(128, 128, batch_first=True)
  (dropout): Dropout(p=0.2, inplace=False)
)
```

Decoder

Embedding layer input size

Attention Mechanism (a feed-forward neural network)

GRU with hidden size of 256

Dropout layer

```
AttentionDecoder(
  (embedding): Embedding(61973, 128)
  (simpleAttention): BahadanuAttention(
    (Wa): Linear(in_features=128, out_features=128, bias=True)
    (Ua): Linear(in_features=128, out_features=128, bias=True)
    (Va): Linear(in_features=128, out_features=1, bias=True) )
  (rnn): GRU(256, 128, batch_first=True)
  (output): Linear(in_features=128, out_features=61973, bias=True)
  (dropout): Dropout(p=0.1, inplace=False)
)
```

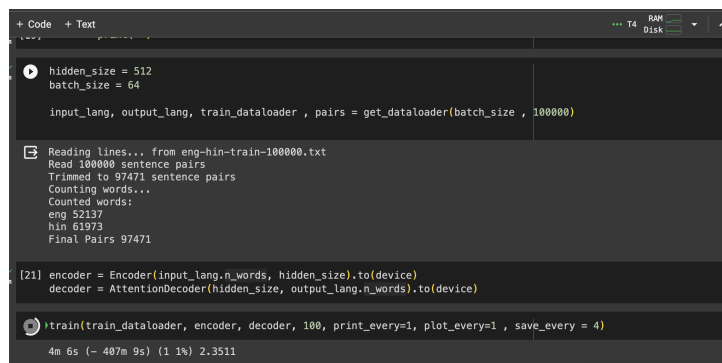
Training :- Used the same data with Google Colab GPU

Time taken - 150m 30s (see Fig. 5)

```
140m 2s (- 10m 32s) (93 93%) 0.2471
141m 32s (- 9m 2s) (94 94%) 0.2460
143m 2s (- 7m 31s) (95 95%) 0.2434
144m 32s (- 6m 1s) (96 96%) 0.2414
Saving ...
Saved Encoder at encoderHindi-iitb-with-attn-100k-gru-data-nonrev-96-epoch.pt
Saved Decoder at decoderHindi-iitb-with-attn-100k-gru-data-nonrev-96-epoch.pt
146m 1s (- 4m 30s) (97 97%) 0.2397
147m 31s (- 3m 0s) (98 98%) 0.2365
149m 1s (- 1m 30s) (99 99%) 0.2349
150m 30s (- 0m 0s) (100 100%) 0.2330
Saving ...
Saved Encoder at encoderHindi-iitb-with-attn-100k-gru-data-nonrev-100-epoch.pt
Saved Decoder at decoderHindi-iitb-with-attn-100k-gru-data-nonrev-100-epoch.pt
```

Figure 5: Training with Attention

We also tried training the models by increasing the hidden size from 128 to 512 but the training time increase to approx 410 mins



```

hidden_size = 512
batch_size = 64

input_lang, output_lang, train_data_loader, pairs = get_data_loader(batch_size, 100000)

Reading lines... from eng-hin-train-100000.txt
Read 100000 sentence pairs
Trimmed to 97471 sentence pairs
Counting words...
Counted words:
eng 52137
hin 61973
Final Pairs 97471

[21] encoder = Encoder(input_lang.n_words, hidden_size).to(device)
      decoder = AttentionDecoder(hidden_size, output_lang.n_words).to(device)

train(train_data_loader, encoder, decoder, 100, print_every=1, plot_every=1, save_every=4)

4m 6s (- 407m 9s) (1.1%) 2.3511

```

Figure 6: Training by increasing hidden size to 512

Some inferences for this design

It was noticed that the model was learning properly (relative to simple encoder-decoder model) on the same dataset but still did not produce very good translations.

It learned long and short translations and it also learned (soft-)alignments as proposed by Bahadanu et. al.

Some of the inferences made above still holds (data, model simplicity)

Preliminary Results

Some Translations from Vanilla Encoder-Decoder Model

Input Sentence :: this is provided by recognition or pratyabhijna .
 Actual Translated Sentence :: यह विज्ञानकला अथवा कवलय की अवस्था है।
 Translated Sentence :: यह विज्ञानकला अथवा कवलय की गई लचीली होती है। <EOS>

Input Sentence :: select the image sub - format
 Actual Translated Sentence :: छवि की सब-फॉर्मट को चुन
 Translated Sentence :: छवि की अनमति द <EOS>

Input Sentence :: inflammation of the spinal cord and its enveloping arachnoid and pia mater .
 Actual Translated Sentence :: रीढ़ की हड्डी और उसके चारों ओर बाल आरकनाइड या पिया मटर की जलना
 Translated Sentence :: रीढ़ की हड्डी और उसके चारों ओर स ढ़क की तरफ पयास (जानवरों की भी)

Input Sentence :: demolition
 Actual Translated Sentence :: विघात
 Translated Sentence :: लगाना <EOS>

Input Sentence :: background color of tasks that are due today, in "#rrggbb" format .
 Actual Translated Sentence :: कार्य का पष्ठभूमि रंग जो आज किया जाना है, "#rrggbb" परारप म .
 Translated Sentence :: कार्य का पष्ठभूमि रंग जो आज किया जाना है, "#rrggbb" परारप म . <EOS>

Input Sentence :: and raise strong mansions as if you were to live forever ?
 Actual Translated Sentence :: और भव्य महल बनात रहोग, मानो तमह सदव रहना ह ?
 Translated Sentence :: और भव्य महल बनात रहोग, मानो तमह सदव रहना ह ? <EOS>

Input Sentence :: "then bring forth your book, if you are truthful !"
 Actual Translated Sentence :: तो लाओ अपनी किताब, यदि तम सचच हो
 Translated Sentence :: तो लाओ अपनी किताब, यदि तम सचच हो <EOS>

Input Sentence :: national institute of technology (deemed university) (external website that opens in a new window)
 Actual Translated Sentence :: राष्ट्रीय प्रौद्योगिकी संस्थान (मानद विश्वविद्यालय) (बाहरी वेबसाइट जो एक नई विंडो में खलती है)
 Translated Sentence :: राष्ट्रीय प्रौद्योगिकी संस्थान (मानद विश्वविद्यालय) (बाहरी वेबसाइट जो एक नई विंडो में खलती है) <EOS>

Some Translations from the Attention Encoder-Decoder Model

Input Sentence :: unconfused
 Actual Translated Sentence :: अभरात
 Translated Sentence :: अभरात <EOS>

Input Sentence :: _ manage favorites
 Actual Translated Sentence :: पसदीदा परबधित कर (_ m)
 Translated Sentence :: पसदीदा परबधन कर (_ e) <EOS>

Input Sentence :: life style depends upon occupation .
 Actual Translated Sentence :: जीवन शली वयवसाय पर निरभर करती ह।
 Translated Sentence :: जीवन शली वयवसाय पर निरभर सीध समरथित करता ह। <EOS>

Input Sentence :: our strengthened capability adds to our sense of responsibility .
 Actual Translated Sentence :: हमारी सदढ कषमता, हमार उत्तरदायितव की भावना को बढाती ह।
 Translated Sentence :: हमारी सदढ कषमता, हमार उत्तरदायितव की भावना को बढाती ह। <EOS>

Input Sentence :: education beats the beauty and the youth .
 Actual Translated Sentence :: शिक्षा सदरता और यौवन को भी मात दती ह।
 Translated Sentence :: शिक्षा सदरता और यौवन को भी मात दती ह। <EOS>

Input Sentence :: they would hardly see each oilier throughout the day .
 Actual Translated Sentence :: दिन भर म मलाकात भी तो कम ही होती ह।
 Translated Sentence :: दिन ही कभी लिखित बयान परतिदिन आदि लखन सधरण बना दिया जाता ह। <EOS>

Input Sentence :: he certainly has a strong love for wealth and riches .
 Actual Translated Sentence :: और निशचय ही वह धन क मोह म बडा दढ ह
 Translated Sentence :: वह वयकति न माल ही माल ही भोजन और भोजन क लिए लोह का परलखन

Input Sentence :: village health guide gives education about health and health problems .
 Actual Translated Sentence :: गरामीण स्वास्थय मारगदर्शक स्वास्थय तथा स्वास्थय समस्याओ क बार म शिक्षा परदान करता ह
 Translated Sentence :: गरामीण स्वास्थय मारगदर्शक स्वास्थय तथा स्वास्थय समस्याओ क बार म शिक्षा परदान करता ह। <EOS>

Input Sentence :: text of pm's inaugural address at india global week 2020
 Actual Translated Sentence :: इडिया ग्लोबल वीक 2020 म परधानमंतरी क उदघाटन भाषण का मलपाठ
 Translated Sentence :: परधानमंतरी कौल पर आय खली तिथि स परधानमंतरी न धवज ऊचा करनवाली जजा <EOS>

Road map for remaining project works

1. Model Refinement and optimization
2. Incorporating Pre-trained Transformers
3. Evaluation Metrics and Comparative Analysis
4. Error Analysis
5. User Interface and Deployment