MACHINE TRANSLATION
(English to Hindi)

Team Members

Yash Sunil Sadhwan -

Swapnil Bhattacharyya -
Ujjwal Sharma -
Saral Sureka -
Trivikram Vidya Umanath -
Sneha Oram -

TASK DESCRIPTION

Problem Statement - Language divergence in our country is rich, but hinders inclusivity among its people. Machine translation still remains a challenge even after several decades of its conceptualization.

Proposed solution approach - Neural Machine Translation (NMT) for translating the English language to the Hindi language using three architecture combinations to optimize translation accuracy.

CHALLENGES ADDRESSED

1. Base models(LSTM/GRU model) took long time for computing output.
2. Attention based model demanded of higher compute power.
3. The transformer model in itself took hours to be trained even with GPU compute.
4. We were not able to train on GPU1 server so we trained parallelly in Google Colab for 2 to 3 hrs.
5. The transformer model also had a larger size which made it difficult for training and deployment.

OUTLINE OF METHOD

Three architectures were used for modeling -

1. Vanilla Encoder Decoder model - LSTM/GRU without attention
    a. Dataset was trimmed to 97471 sentence pairs
2. Encoder Decoder model with Attention Decoder
    a. Dataset was trimmed to 97471 sentence pairs
3. Transformer model
    a. Trained different models with 150k, 200k, 250k and 300k sentence pair parallel corpus
    b. After preprocessing the datasets were trimmed to ~90k , ~135k , ~169k and ~200k sentence pairs

GitHub Repository - https://github.com/ujjwalsharmaIITB/FML-Project-NMT-En-Hi.git

EXPERIMENT DETAILS AND MAIN RESULTS

Dataset - IIT Bombay Hindi English Corpus

Data Preprocessing - Building vocabulary, converting sentence to tensors, removing code-mixed sentences, null pairs, punctuations, lower casing, changing encoding of sentence, maintaining maximum sentence length.

Modelling -

1. Vanilla Encoder Decoder Model -
   - Encoder - LSTM/GRU with hidden size of 128, Embedding(52137, 128), GRU(128, 128, batch first=True)
   - Decoder - LSTM/GRU with hidden size of 128, Embedding(61973, 128), GRU(128, 128, batch first=True)
   - Time Taken: 134 mins

2. Encoder Decoder model with Attention Decoder -

- Encoder - LSTM/GRU with hidden size of 128, Embedding(52137, 128), GRU(128, 128, batch first=True)
- Decoder - LSTM/GRU with hidden size of 256, Embedding(61973, 128), Attention Mechanism(Bahdanau Attention), GRU(256, 128, batch first=True)
- Time taken - 150m 30s

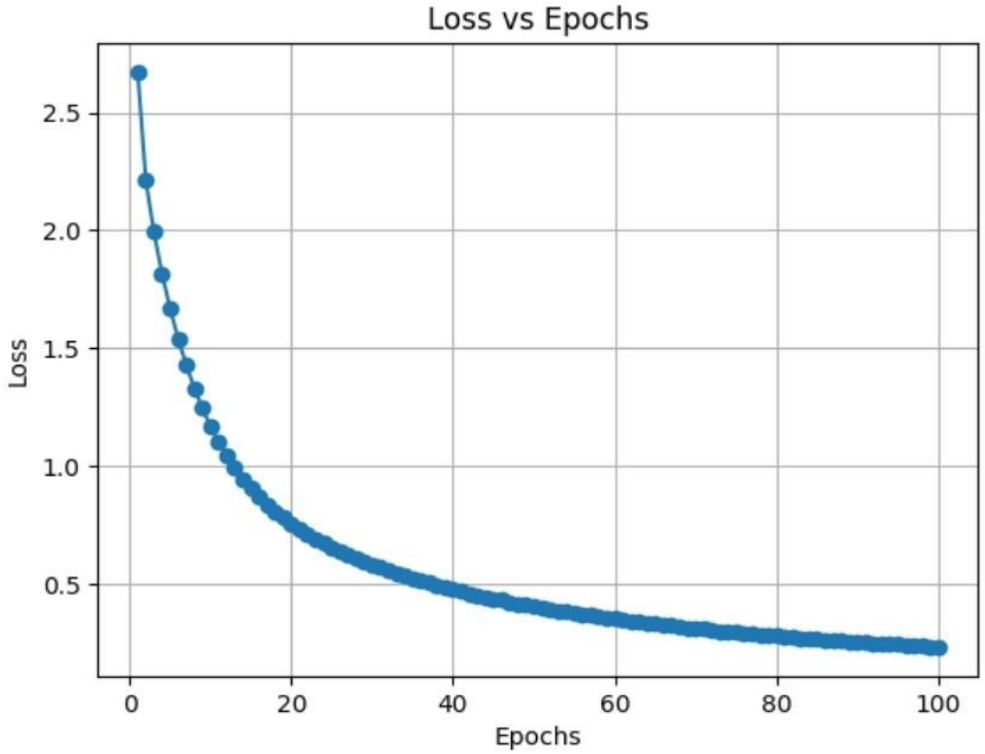
3. Transformer model (with 150k, 200k, 250k and 300k sentence pair)

- Embedding size=512
- Number of attention heads=8
- One encoder and one decoder
- dropout=0.10

Note :- We were able to train only 20 epochs for the transformers in

Google Colab For 200k ~ 97 mins ,250k ~ 132 mins , 300k ~ 178 mins

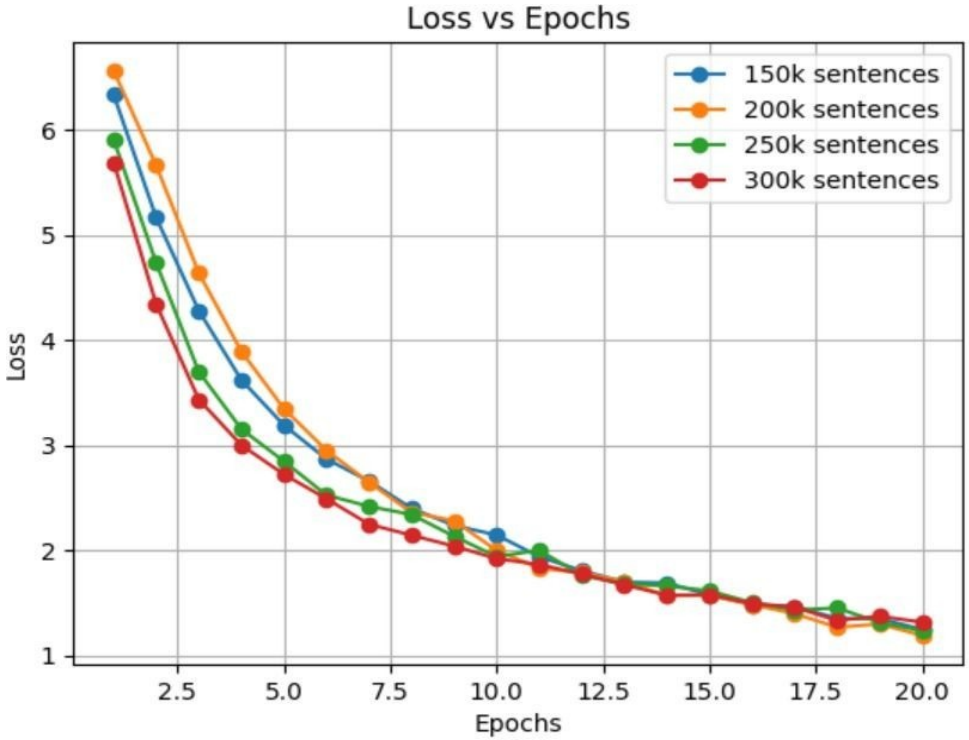
Loss (training) functions for sequence to sequence models



Loss function without attention

Loss function with attention

Loss function with Transformer model (with varying sentence pairs)



Translations from attention-based models

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

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

Translation from transformer (200k training instances)

English: <SOS> now jeevan dutt could speak out to the world without hesitation and say that it was a fact <EOS> <PAD> <PAD>

Actual: <SOS> जीवनदत्त ऊंची आवाज से घोषित कर सकते हैं कि यह सत्य है सत्य है। <EOS> <PAD> <PAD> <PAD> <PAD> <PAD> <PAD>

Predicted: जीवनदत्त ऊंची आवाज fमल सकता है और तब हमें उस समय दुfनया में जीत सकता है।

BLEU scores

Model/n-gram	1-gram	2-gram	n-gram
Transformer (200k)	0.15(appr.)	0.06 (appr.)	0.01 (appr.)
Transformer (250k)	0.166 (appr.)	0.0736 (appr.)	0.0125 (appr.)
Transformer (300k)	0.168 (appr.)	0.0749 (appr.)	0.0125 (appr.)

Deployment and Interface

We also made a interface (web application).

You can select from different models to fetch the translation

RELATED WORKS

1) J. Nair, K. A. Krishnan and R. Deetha, "An efficient English to Hindi machine translation system using hybrid mechanism," 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, India, 2016, pp. 2109-2113, doi: 10.1109/ICACCI.2016.7732363.
2) S. Saini and V. Sahula, "Neural Machine Translation for English to Hindi," 2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP), Kota Kinabalu, Malaysia, 2018, pp. 1-6, doi: 10.1109/INFRKM.2018.8464781.

CONCLUSION

Due to limited data and lack of huge compute support for machine translation, using RNNs and transformers, our models performed well enough on training sentences while it performed quite inaccurately on different sentences.

We noticed that the RNN models that were trained by “Teacher Forcing” were learning the correct english words but diverged a lot, this is somewhat controlled by incorporating attention mechanisms

Transformers performed much better than the RNN models on the same dataset and we were able to train on more data while having approximately the same training time.

Overall we tried to perform a study of how different models perform on same datasets and similar prompts.

We also deployed the model (made a web application for translation) using Flask .

WORK SPLIT UP AMONG TEAM MEMBERS

There was no strict division of tasks everyone contributed almost equally in training and reporting.The following areas are where contributions were more by the members:-

1) Coding, training models and analysis:-Ujjwal Sharma, Swapnil Bhattacharyya, Saral Sureka and Yash Sadhwan.
2) Reports, PPT, code survey, paper reading and team management:-Sneha Oram and Trivikram Vidhya Umanath

REFERENCES / LIBRARIES

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. Library Used
    a. Pytorch
    b. Nltk
    c. Matplotlib
    d. Datasets (for downloading IITB corpus)
    e. Flask (for deployment and ApiS)

REFERENCES / LIBRARIES (contd.)

1. https://arxiv.org/abs/1706.03762
2. https://www.youtube.com/watch?v=iDulhoQ2pro
3. https://www.youtube.com/watch?v=TQQlZhbC5ps&t=636s
4. https://pytorch.org/tutorials/beginner/translation_transform er.html