Efficient Neural Machine Translation for Indian Languages

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science in ***Computer Science and Engineering*** *by Research*

by

Vikrant Goyal
201502040
vikrant.goyal@research.iiit.ac.in

International Institute of Information Technology
Hyderabad - 500 032, India
October 2020

International Institute of Information Technology
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis titled "Efficient Neural Machine Translation for Indian Languages" by Vikrant Goyal has been carried out under my supervision and is not submitted elsewhere for a degree.

Date	Advisor: Dr. Dipti Misra Sharma

Dedicated to my

Mother

Smt. Sangeeta Goyal

&

Papa

Shri. Kewal Krishan Goyal

&&

Elder Brother

Karan Goyal

Acknowledgements

It has been almost 3 years since I started working in LTRC, IIIT Hyderabad. From starting as a student who was clueless about what research is, to submitting my MS thesis now, I feel I've learned a lot during this time. I would like to offer my gratitude to everyone who has been a part of my journey. First and foremost, I would like to express my gratitude to my advisor Prof. Dipti Misra Sharma for showing the faith and confidence in me and guiding me during the course of my work. She contributed not only towards my academic growth, but towards an all round development across all areas of my life. She was there to support me at all stages during my tenure as a research student at IIIT Hyderabad.

I owe my thanks to a number of people, each of whom contributed in their own way towards the completion of this work. A special mention to Pruthwik Mishra , Vandan Mujadia and Saumitra Yadav who inspired me a lot and gave me due direction with their experience.

My tenure here has ignited my interest in research. Apart from the academics, I have truly enjoyed my stay at IIIT Hyderabad thanks to the friends I made - Ishan, Vaibhav, Akhilesh, Paawan, Akhil, Malani and Sharique. I am grateful to my seniors for their academic guidance.

Most importantly, I am very thankful to my family for their constant support and motivation.

Abstract

Neural Machine Translation (NMT) is a rapidly advancing MT paradigm and has shown promising results for many language pairs, especially in large training data scenarios. But Indian to English language Machine Translation is a challenging problem, owing to multiple factors including the structural and morphological difference, in addition to the lack of sufficient parallel training data, thereby demanding efficient strategies to improve the translation quality of the NMT systems.

Although NMT is a promising approach, it still lacks the ability of modeling deeper semantic and syntactic aspects of the language. In machine translation with a low-resource setting, resolving data sparseness and semantic ambiguity problems can help improve its performance. In this thesis, we investigate utilizing extra syntactic and semantic linguistic factors in the context of the NMT framework for a low resource language pair i.e. Hindi-English. We propose a new architecture to incorporate explicit linguistic input features into the state-of-the-art Transformer network and demonstrate considerable performance improvements.

Despite the massive success brought by neural machine translation, it has been noticed that the vanilla NMT often lags behind conventional machine translation systems, such as statistical phrase-based translation systems, for low-resource language pairs. In the past few years, various approaches have been proposed to address this issue but not much work has been done for Indian languages in this context. In this thesis, we also present our efforts towards building efficient Neural Machine Translation systems between Indian languages (specifically Indo-Aryan languages) and English via exploring the effectiveness of Multilingual Learning and Transfer Learning. We describe techniques to leverage the language relatedness among Indo-Aryan languages to improve the translation quality for individual language pairs. We also present our new approach Multilingual Transfer Learning which outperforms the aforementioned techniques.

Neural MT models are generally trained using a Maximum Likelihood Estimation (MLE) objective and are tested with sequence level evaluation metrics such as BLEU. To address this inconsistency issue, Reinforcement Learning (RL) methods have been adopted to directly optimize sequence-level objectives. In this thesis, we also present an approach for training Neural Machine Translation systems using Advantage Actor-Critic method from Reinforcement Learning. Our approach directly optimizes the model parameters with respect to the task-

specific scores, unlike conventional maximum likelihood estimation and is fit for problems with low resource settings, large action space & delayed rewards. We also demonstrate experiments to leverage our approach to further boost the performance of NMT systems using source & target monolingual data for a low resource language pair like Hindi-English.

Contents

Chapter Page

1 Introduction . . . 1
1.1 Problem Overview . . . 2
1.2 Contributions . . . 3
1.3 Organization of the Thesis . . . 4

2 Preliminary - Neural Machine Translation . . . 6
2.1 Introduction . . . 6
2.2 Attention-based encoder-decoder framework . . . 7
2.3 The Transformer Architecture . . . 7
2.4 Summary . . . 9

3 Linguistically Informed Hindi-English Neural Machine Translation . . . 10
3.1 Introduction . . . 10
3.2 Adding Input Linguistic Features . . . 12
3.3 Linguistic Input Features . . . 12
3.3.1 Lemma . . . 13
3.3.2 Morphological Features . . . 13
3.3.3 POS Tags . . . 13
3.3.4 Using Word-level Features in the Subword Model . . . 13
3.4 Experimental Settings . . . 14
3.4.1 Dataset . . . 14
3.4.2 Data Processing . . . 14
3.4.3 Training Details . . . 15
3.5 Results and Discussion . . . 15
3.6 Related Work . . . 18
3.7 Summary . . . 18

4 Efficient Neural Machine Translation for Low Resource Languages via Exploiting Related Languages . . . 19
4.1 Introduction . . . 19
4.2 Methodology . . . 21
4.2.1 Language Relatedness . . . 22
4.2.2 Unified Transliteration and Subword Segmentation . . . 22
4.2.3 Multilingual Learning for NMT . . . 23
4.2.4 Transfer Learning for NMT . . . 23

4.2.5 Multilingual Transfer Learning for NMT . . . 24
4.3 Experimental Settings . . . 25
4.3.1 Dataset . . . 25
4.3.2 Data Processing . . . 25
4.3.3 Training Details . . . 26
4.4 Results and Discussion . . . 26
4.5 Summary . . . 28

5 The IIIT-H Gujarati-English Machine Translation system for WMT19 . . . 31
5.1 Introduction . . . 31
5.2 Multilingual Neural Machine Translation . . . 32
5.3 Experimental setup . . . 32
5.3.1 Dataset . . . 32
5.3.2 Data Processing . . . 33
5.3.3 Subword Segmentation for NMT . . . 33
5.3.4 Script Conversion . . . 33
5.3.5 Training Details . . . 34
5.4 Results . . . 34
5.5 Additional Transformer Experiments . . . 34
5.6 Summary . . . 36

6 LTRC-MT Simple & Effective Hindi-English Neural Machine Translation Systems at WAT 2019 . . . 37
6.1 Introduction . . . 37
6.1.1 Subword Segmentation for NMT . . . 38
6.1.2 Synthetic Training Data . . . 38
6.2 Experimental Setup . . . 38
6.2.1 Dataset . . . 38
6.2.2 Data Processing . . . 39
6.2.3 Training Details . . . 39
6.3 Results and Discussion . . . 39
6.4 Summary . . . 40

7 A2C-NMT: A Reinforcement Learning Approach for Neural Machine Translation . . . 41
7.1 Introduction . . . 41
7.2 Reinforcement Learning architecture of A2C-NMT . . . 42
7.2.1 Formulation of NMT model as an MDP . . . 42
7.2.2 Policy Gradient Method for our NMT model . . . 43
7.2.3 Advantage Actor-Critic . . . 43
7.3 Experiments . . . 44
7.4 Results and Discussion . . . 46
7.5 Related Work . . . 47
7.6 Summary . . . 48

8 Conclusions Future Work . . . 49

Bibliography . . . 52

List of Figures

Figure Page

2.1 Transformer model architecture from [1] 8

4.1 Model architecture of Multilingual Transfer Learning approach. Single initialized NMT (Figure 1(a)) fine-tunes the child model only with a single high-resource parent model [2]. In the main architecture (Figure 1(b)) of our proposed approach, Multilingual Transfer Learning, a multilingual NMT model is first constructed with various language pairs that are semantically, syntactically similar, and share many words with the child model. This multilingual parent model, which may or may not contain the child language pair, is then finetuned on the low resource language pair of interest (i.e. child model). 21

4.2 Our pipeline for building Multilingual NMT models for Indian languages. 22

4.3 Our pipeline for building Transfer Learning models for Indian languages. 24

List of Tables

Table Page

3.1 Statistics of our processed parallel data. . . . 14
3.2 Size of embedding layer of linguistic features, in system that includes all features, and contrastive experiments that add a single feature over the baseline. The embedding layer size of the word or subword feature is set to bring the total size to 512. . . . 15
3.3 Contrastive experiments for a word based Hindi-English Transformer model with individual linguistic features. . . . 16
3.4 Contrastive experiments for a subword Hindi-English Transformer model with individual linguistic features. . . . 16
3.5 Ablation study showing the effect of usage of linguistic features in a subword model with varying data size. . . . 17
3.6 Translation examples illustrating the effect of adding linguistic input features. . . . 17

4.1 Statistics of our cleaned & processed parallel data, where XX is: Gujarati, Marathi, Bengali or Punjabi . . . 25
4.2 BLEU scores of the contrastive experiments for Indian Language to English translation (XX to EN). . . . 26
4.3 BLEU scores of the contrastive experiments for English to Indian Language translation (EN to XX). . . . 27
4.4 Punjabi-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system. . . . 28
4.5 Bengali-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system. . . . 28
4.6 Gujarati-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system. . . . 29
4.7 Marathi-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system. . . . 30

5.1 Statistics of our processed parallel data. . . . 33
5.2 WMT19 evaluation of our systems . . . 34
5.3 Our Transformer models vs other systems at WMT19 . . . 35

5.4 Gujarati-English Translation examples illustrating the quality difference between our baseline and Multilingual models. 35

6.1 Statistics of our processed parallel data. 39
6.2 This table describes the results of WAT 2019 evaluation of our submitted systems & compared with the best system submissions of WAT 2019 & the previous year. 'BT' stands for backtranslation. 40
6.3 Hindi-English Translation examples illustrating the quality of our baseline and best performing models (backtranslation) submitted at WAT@EMLP-IJCNLP 2019 shared task. 40

7.1 Comparison with previous work on En→Fr translation task. 46
7.2 Comparison with previous work on IWSLT2014 German-English translation task. 47
7.3 Comparison with previous work on WAT2017 Hi→En translation task. 47

Chapter 1

Introduction

India is a big and a diverse country where people from different regions speak in different languages. Information exchange and communication between people is necessary not only for business purposes but also for the people to share their opinions, feelings, thoughts and facts with each other. We need some effective approaches which do this job with as little human effort as possible because it is not feasible to have human translators everywhere. Machine Translation (MT) is a sub-field of Computational Linguistics which enables automatic translation of sentences or documents from one natural language to another. MT systems aim to generate quality translations which are syntactically correct in the target language and are semantically equivalent to the source sentence.

Research work in MT dates back to as early as the 1950s [3, 4], and has progressed rapidly since the 1990s because of the availability of large bilingual and multi-lingual text corpora and also due to the increased computing power and storage. In the past decades multiple approaches for solving complex MT problems have been suggested: rule-based translation [5, 6], knowledge-based translation [7], corpus-based translation [8] and hybrid translation [9]. Each of these approaches has its own pros and cons. Rule-based Machine Translation (RBMT) approach rely on linguistic rules and bilingual dictionaries for each language pair wherein the rules capture the syntactic and semantic properties of a language. Rules are written with linguistic knowledge gathered from linguists. Statistical Machine Translation (SMT) approach (which can be sub-categorized under corpus-based translation) uses a statistical model to generate translations and is based on the analysis of bilingual corpus. The SMT approach has seen an increasing interest in the past because of different factors, which range from the growing availability of parallel data, together with the increasing computational performance, to the successful results achieved in several evaluation campaigns, which are proved to be as good or even better than results of system following the rule-based approach. The most important benefit of SMT over RBMT is that it does not require manual development of linguistic rules, which is quite costly. Some of the notable works on SMT are [10, 11, 12], where the authors have dived deep into various challenges, working principles and possible improvements. SMT has shown good results

for many language pairs and is responsible for the popularity of MT among general public in the past.

While SMT was successfully deployed in many commercial systems in the past, it does not work very well and suffers from the following two major drawbacks. First, translation decisions are locally determined as we translate phrase-by-phrase and long-distance dependencies are often ignored. As more and more features are added to the loglinear SMT framework such as in the past MT systems [13, 14, 15], the entire MT pipeline becomes increasingly complex. Many different components need to be tuned separately, e.g., translation models, language models, reordering models, etc., which makes it difficult to combine them together and to innovate.

Neural Machine Translation (NMT) is the most recent MT approach that addresses the aforementioned problems posed by SMT. In NMT, a single big neural network (with millions of artificial neurons) often consisting of an encoder and a decoder is designed to model the entire MT process [16, 17, 18, 19]. NMT requires minimal domain knowledge, just a parallel corpus of source and target sentence pairs, similar to SMT, but with far less preprocessing steps before a translation model can be built. The most appealing feature of NMT is that it can be trained end-to-end directly from the learning objective; hence, eliminating the problem of having to learn multiple components in SMT systems. NMT generates more fluent translations as compared to phrase based SMT systems especially on lexically rich texts. NMT has been reported to significantly improve over SMT both in automatic metrics and human evaluation [20] and has become the dominant paradigm to machine translation.

Inspite of the advantages of NMT over conventional phrase-based SMT techniques, it is still not the perfect solution and has many limitations. For example, it requires a large parallel corpus to be effective, and is known to fail when the training data is not big enough [21]. NMT also lacks the ability to model deeper semantic and syntactic aspects of the language which is major problem for languages with rich morphology. NMT models are usually trained using Maximum Likelihood Objective (MLE) function and this token-level objective function during training is inconsistent with the sequence level evaluation metrics such as BLEU [22] as shown in [23]. We address these challenges in this thesis.

1.1 Problem Overview

Big countries such as India and China have several languages which change by regions. For instance, India has 22 scheduled languages (e.g., Hindi, Punjabi, and Telugu) and several hundreds of unofficial local languages. Languages spoken in India belong to several language families, the major ones being the Indo-Aryan, Dravidian, Austroasiatic and Sino-Tibetan language families. Also, Indian languages have a prominent presence in the languages spoken across the world, both in terms of the linguistic characteristics as well as the socio-cultural aspects. With the presence of such a large number of languages with diverse characteristics,

communication across different linguistic groups can be facilitated to a great extent with the help of the technology of Machine Translation. Owing to the above factors, there is a lot of scope for work which can be done in the direction of Indian language MT - a field with many potential applications in domains like education, business, tourism, communication, government and so on.

However, there are many challenges in machine translation for English to Indian languages. For instance, (i) the extremely low size of parallel corpora and (ii) differences amongst languages, mainly the morphological richness and word order differences due to syntactical divergence are two of the major challenges. Moreover, Indian languages such as Hindi differ from English in word order as well as in morphological complexity. For example, English has Subject-Verb-Object (SVO) whereas Hindi has Subject-Object-Verb (SOV) word order. The parallel corpora available for English and Indian languages is also very low. While syntactic differences contribute to difficulties of translation models, morphological differences contribute to data sparsity.

Though much work is being done on machine translation for foreign and Indian languages, it is limited to conventional machine translation techniques only [24, 25, 26, 27, 28, 29]. Therefore, efficient Neural Machine Translation techniques are needed to address the aforementioned challenges in the Indian language to English translation.

1.2 Contributions

The major contributions of this thesis are summarized as below:

1. In order to alleviate problem of data sparsity in Neural Machine Translation of Hindi to English that is caused due the morphological and syntactical differences between Hindi & English, we propose a new method to employ additional linguistic knowledge which is encoded by different phenomena depicted by Hindi. We generalize the embedding layer of the state-of-the-art Transformer model to incorporate linguistic features like POS tag, lemma and morph features. We compare the results obtained on incorporating this linguistic knowledge with the baseline NMT systems and demonstrate significant performance improvements.

2. The condition of large parallel corpora is not met for Indian-English language pairs which is necessary for efficient NMT. So we present our efforts towards building efficient Neural Machine Translation systems between Indian languages (specifically the Indo-Aryan languages) and English via efficiently exploiting parallel data from the related languages. We propose a technique called Unified Transliteration and Subword Segmentation to leverage language similarity while using parallel data from related language pairs. We also propose a new technique called Multilingual Transfer Learning to leverage parallel data from

multiple related languages to assist translation for low resource language pair of interest in the context of Transfer Learning. Our experiments demonstrate an overall average improvement of 5 BLEU points over the standard Transformer based NMT baseline.

3. In this thesis, we also propose a new approach to leverage a Reinforcement Learning technique to boost the performance of standard Neural Machine Translation systems. Our approach also outperforms some previous RL baselines.

1.3 Organization of the Thesis

The chapters in this thesis are designed in such a way so that a reader can skip to a particular chapter and still be able to comprehend the material in a standalone fashion. This is for the benefit of the reader and I hope this will aid in increasing the interest-factor of the thesis since the reader can now read it in multiple sittings, without requiring significant re-reading of previous chapters.

The rest of the thesis is organized as follows:

Chapter 2 presents a concise description of the Neural Machine Translation models used in this work.

Chapter 3 talks about our novel contribution of building a Linguistically Informed NMT system for Hindi-English leveraging the state-of-the-art Transformer Model with explicit linguistic input features to reduce data sparsity. Our linguistically informed Transformer architecture supports arbitrary no. of linguistic features on the source side of the NMT model.

Chapter 4 seeks to leverage parallel data of multilple related languages to assist translation of low resource languages (Indo-Aryan). We present a simple technique to leverage language similarity of related languages to enhance the translation quality. We also propose a new technique called Multilingual Transfer Learning to leverage multiple related language pairs together to provide better knowledge transfer to low resource languages.

Chapter 5 presents our work on building efficient NMT system for Gujarati-English language pair as part of WMT19 shared task. We leverage training data from Hindi-English language pair to assist translation for Gujarati to English via developing a Multilingual Translation model. Experiments reveal that our approach helps in significant BLEU improvements upto 11.5 over the baseline NMT.

Chapter 6 presents our work on Hindi-English NMT using Recurrent Neural Networks and Transformer architecture. The baseline NMT do not yield acceptable translation quality due

to limited training data. However, the use of synthetic parallel data (generated using back translation, based on an NMT baseline) significantly improves translation quality. Our best performing translation system ranked as the runner-up amongst all the systems that participated in the Hindi to English translation task as a part of Workshop on Asian Translation (WAT) @ EMNLP-IJCNLP 2019.

Chapter 7 presents an approach for training Neural Machine Translation systems using Advantage Actor-Critic method from Reinforcement Learning. We also demonstrate experiments to leverage our approach to further boost the performance of NMT systems using source and target monolingual data for a low resource language pair. On standard translation tasks, our approach outperforms some strong RL baselines for NMT.

Chapter 8 presents the conclusions of the work in this thesis and the scope for future work on this area.

Chapter 2

Preliminary - Neural Machine Translation

In this chapter, we lay out a concise description of the theoretical background of Neural Machine Translation needed to understand this thesis in depth. The thesis proposes some efficient techniques to improve the use of Neural Machine Translation for the task of Indian Language MT. Neural Machine Translation (NMT) is a novel approach to MT which utilizes deep neural networks to generate end-to-end translation. The theoretical background behind NMT is described in the following sections. We begin with a brief introduction to NMT, followed by the description of attention-based encoder-decoder architecture for NMT and end with a description of the self-attention mechanism - state-of-the-art approach to NMT.

2.1 Introduction

Artificial Neural Networks are an inevitable building block for recent advances using deep learning for Natural Language Processing. Simplistically, a neural network is a program which is designed to work in a manner similar to that of the human brain. When we employ a single artificial neural network to build a model for the task of end-to-end Machine Translation, the resulting approach is termed as Neural Machine Translation.

NMT is a seq2seq model which takes a sequence of tokens in one language as input and returns a sequence of tokens into a different language as output. A basic NMT model consists of two main components - the encoder and the decoder. Typically, the encoder takes in the input sequence in one-hot vector format and maps it into a higher dimensional vector representation. The final hidden state of the encoder which is also known as thought vector, stores the meaning of the source sentence and is subsequently used by the decoder to generate the translation in target language.

2.2 Attention-based encoder-decoder framework

In this architecture, the NMT model consists of an encoder and a decoder, each of which is a Recurrent Neural Network (RNN) as described in [30]. The model directly estimates the posterior distribution $P_\theta(y|x)$ of translating a source sentence $x = (x_1, .., x_n)$ to a target sentence $y = (y_1, .., y_m)$ as:

$$P_\theta(y|x) = \prod_{t=1}^{m} P_\theta(y_t|y_1, y_2, .., y_{t-1}, x) \tag{2.1}$$

Each of the local posterior distribution $P(y_t|y_{1,2}, .., y_{t-1}, x)$ is modeled as a multinomial distribution over the target language vocabulary which is represented as a linear transformation followed by a softmax function on the decoder's output vector $\tilde{h}_t^{dec}$:

$$c_t = AttentionFunction(h_{1:n}^{enc}; h_t^{dec}) \tag{2.2}$$

$$\tilde{h}_t^{dec} = tanh(W_o[h_t^{dec}; c_t]) \tag{2.3}$$

$$P(y|y_1, y_2, .., y_{t-1}, x) = softmax(W_s\tilde{h}_t^{dec}; \tau) \tag{2.4}$$

where c_t is the context vector, h^{enc} and h^{dec} are the hidden vectors generated by the encoder and decoder respectively, AttentionFunction(. , .) is the attention mechanism as shown in [30] and [. ; .] is the concatenation of two vectors.

An RNN encoder first encodes x to a continuous vector, which serves as the initial hidden vector for the decoder and then the decoder performs recursive updates to produce a sequence of hidden vectors by applying the transition function f as:

$$h_t^{dec} = f(h_{t-1}^{dec}, [\tilde{h}_{t-1}^{dec}; e(y_t)]) \tag{2.5}$$

where e(.) is the word embedding operation. Popular choices for mapping f are Long-Short-Term Memory (LSTM) units and Gated Recurrent Units (GRU), the former of which we use in our models.

An NMT model is typically trained under the maximum log-likelihood objective:

$$\max_\theta J(\theta) = \max_\theta \mathrm{E}_{(x,y)\sim D}[\log P_\theta(y|x)] \tag{2.6}$$

where D is the training set. Unless specified, our NMT model uses a bi-directional LSTM as an encoder and a uni-directional LSTM as a decoder with global attention [30] .

2.3 The Transformer Architecture

The Transformer [1] model is the first NMT model relying completely on self-attention mechanism to compute representations of its input and output without using recurrent neural

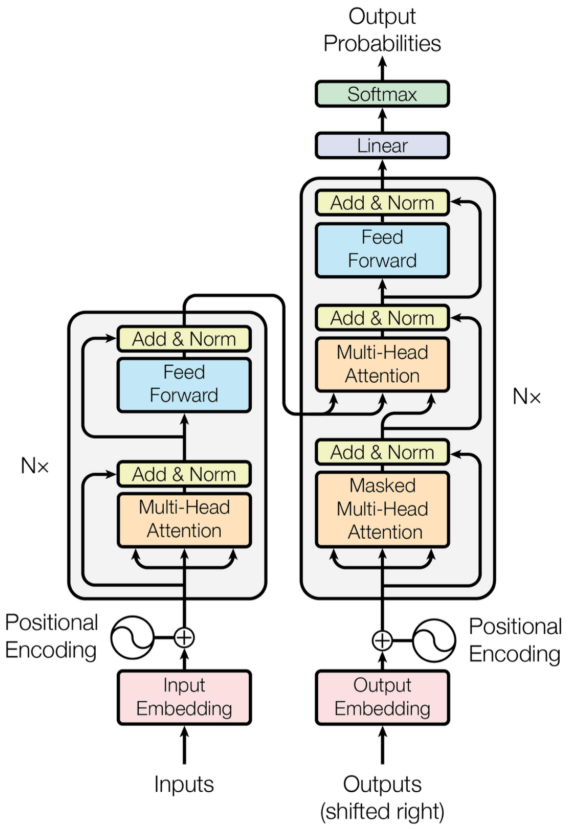

Figure 2.1: Transformer model architecture from [1]

networks (RNN) or convolutional neural networks (CNN).

RNNs read one word at a time, having to perform multiple steps before generating an output that depends on words that are far away. But it has been shown that the more steps required, the harder it is for the network to learn to make these decisions [19]. RNNs being sequential in nature, do not effectively exploit the modern computing devices such as GPUs which rely on parallel processing.

The Transformer is also an encoder-decoder model that was conceived to solve these problems. Without using any recurrent layer, the model takes advantage of the positional embedding as a mechanism to encode order within a sentence. The encoder, typically stacks 6 identical layers, in which each of them makes use of the so called multi-head attention and of a 2 sub-layers feed-forward network, coupled with layer normalization and residual connection (see Figure 2.1). The multi-head attention mechanism computes attention weights, i.e., a softmax distribution, for each word within a sentence, including the word itself. Specifically:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2.7}$$

where the input consists of queries Q and keys K of dimension d_k, and values V of dimension d_v. The queries, keys and values are linearly projected h times, to allow the model to jointly attend to information from different representation, concatenating the result,

$$Multihead(Q, K, V) = Concat(head_1, ..., head_h)W^o \tag{2.8}$$

where,

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{2.9}$$

with parameter matrices $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ and $W_i^o \in \mathbb{R}^{hd_v \times d_{model}}$.

On top of the multi-head attention there is a feed-forward network that consists of two layers with a ReLU activation in between. Each encoder layer takes as input the output of the previous layer, allowing it to attend to all positions of the previous layer.
The decoder has the same architecture as the encoder, stacking 6 identical layers of multi-head attention with feed-forward networks. However, here there are two multi-head attention sub-layers: i) a decoder self-attention and ii) a encoder-decoder attention. The decoder self-attention attends on the previous predictions made step by step, masked by one position. The second multi-head attention performs an attention between the final encoder representation and the decoder representation.

To summarize, the Transformer model consists of three different attentions: i) the encoder self-attention, in which each position attends to all positions in the previous layer, including the position itself, ii) the encoder-decoder attention, in which each position of the decoder attends to all positions in the last encoder layer, and iii) the decoder self-attention, in which each position attends to all previous positions including the current position.

2.4 Summary

In his chapter, we discussed the theory of NMT by briefly describing the LSTM-based attentional encoder-decoder architecture and the state-of-the-art Transformer model. In the next chapters, we try to address the limitations of NMT with a special focus on Indian Languages and use the NMT architectures described in this chapter as the baselines for our experiments.

Chapter 3

Linguistically Informed Hindi-English Neural Machine Translation

Neural Machine Translation (NMT) is the most recent approach to MT and has shown promising results for many language pairs but it still lacks the ability of modeling deeper semantic and syntactic aspects of the language. In this chapter, we propose a method to employ additional linguistic knowledge which is encoded by different phenomena depicted by the source language. We generalize the embedding layer of the state-of-the-art Transformer model to incorporate linguistic features like POS tag, lemma and morph features. We compare the results obtained on incorporating this knowledge with the baseline systems and demonstrate significant performance improvement. We observe that although the Transformer NMT models have a strong efficacy to learn language constructs, the usage of specific features further help in improving the performance.

3.1 Introduction

In recent years, Neural Machine Translation [30, 19, 31, 32, 1] (NMT) has become the most prominent approach to Machine Translation (MT) due to its simplicity, generality and effectiveness. In NMT, a single neural network often consisting of an encoder and a decoder is used to directly maximize the conditional probabilities of target sentences given the source sentences in an end-to-end paradigm. NMT models have been shown to surpass the performance of previously dominant statistical machine translation (SMT) [12] on many well-established translation tasks. Unlike SMT, NMT does not rely on sub-modules and explicit linguistic features in crafting the translation . Instead, it learns the translation knowledge directly from parallel sentences without resorting to additional linguistic analysis.

Although NMT is a promising approach, it still lacks the ability of modeling deeper semantic and syntactic aspects of the language. In machine translation with a low-resource setting,

resolving data sparseness and semantic ambiguity problems can help improve its performance. Addition of explicit linguistic knowledge may be of great benefits to NMT models, potentially reducing language ambiguity and alleviating data sparseness further. Some recent studies have shown that incorporating linguistic features in the NMT model can improve the translation performance [33, 34, 35]. But most of the previous works have shown the effectiveness of usage of linguistic features with the RNN models. However, it is essential to verify whether the strong learning capability of the current state-of-the-art Transformer models make the explicit linguistic features redundant or if they can be easily incorporated to provide further improvements in translation performance.

Also, there is an immense scope in the development of translation systems which cater to the specific characteristics of languages under consideration. Indian languages are not an exception to this, however, they add certain specifications which need to be considered carefully for effective translation. English and Hindi are respectively reported to be the 3rd and 4th largest spoken languages in the world [1] and this fact makes Hindi-English as an ideal language pair for translation studies. But Hindi-English MT is a challenging task because it's a low resource language pair and both the languages belongs to different language families. Hindi is a morphologically rich language and depict unique characteristics, which are significantly different from languages such as English. Some of these characteristics are the relatively free word-order with a tendency towards the Subject-Object-Verb (SOV) construction, a high degree of inflection and usage of reduplication. For example, " बारिश हो रही है। " in Hindi (transliteration: "baarish ho rahii hai.") will be translated to "It is raining." in English. In Hindi "baarish" is the subject but in English it becomes as a verb "rain".

Also when translating between morphologically rich and free word order languages like Hindi and the other end of morphologically less complicated and word order specific languages like English, the well-known issues of missing words and data sparsity arise; and hence affect the accuracy of translation and leads to more out-of-vocabulary (OOV) words.

In this chapter, we present our efforts towards building a Hindi to English Neural Machine Translation system using the state-of-the-art Transformer models via exploiting the explicit linguistic input features on the source side. We generalize the embedding layer of the encoder in the standard Transformer architecture to support the inclusion of arbitrary features, in addition to the baseline token feature, where the token can either be a word or a subword. We add morphological features, part-of-speech (POS) tags and lemma as input features to Hindi-English NMT model.

[1] http://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers

3.2 Adding Input Linguistic Features

Our main innovation over the standard Transformer encoder-decoder architecture is that we represent its encoder input as a combination of features [36].

Let $E \in \mathbb{R}^{m \times K}$ be the word embedding matrix for the standard Transformer encoder with no input features where m is the word embedding size and K is the vocabulary size of the source language. Therefore, the m-dimensional word embedding $e(x_i)$ of the token x_i (one-hot encoded representation i.e. 1-of-K vector) in the input sequence $x = (x_1, x_2, ..., x_n)$ can be written as:

$$e(x_i) = Ex_i \tag{3.1}$$

We generalize this embedding layer to some arbitrary number of features $|F|$:

$$\bar{e}(x_i) = merge(E_j x_{ij}) \tag{3.2}$$

where $E_j \in \mathbb{R}^{m_j \times K_j}$ are the feature embedding matrices with m_j as the feature embedding size and K_j as the vocabulary size of the jth feature. Basically we look up separate embeddings for each feature, which are then merged by some merge function. In this work, we experiment with concatenation of separate embeddings (each with some different embedding sizes) for each feature as the merge operation similar to what was done by [33] in a RNN based attentional NMT system. The length of the final merged vector matches the total embedding size, that is $\sum_{j=1}^{|F|} m_j = m$ and the rest of the model remains unchanged.

3.3 Linguistic Input Features

Our generalized model described in the the previous section supports an arbitrary number of input features, where each of the feature embeddings can also be merged with some merge function other than concatenation. In this work, we focused on a number of well known linguistic features. The main empirical question that we address in this work is if providing linguistic input features to the state-of-the art Transformer model improves the translation quality of Hindi-English neural machine translation systems, or if the information emerges from training encoder-decoder models on raw text, making its inclusion via explicit features redundant. All linguistic features are predicted automatically; we use *Hindi Shallow Parser* [2] to annotate Hindi raw text with the linguistic features. We here discuss the individual features in more detail.

[2] *http://ltrc.iiit.ac.in/showfile.php?filename=downloads /shallow_parser.php*

3.3.1 Lemma

In a normal NMT model each word form is treated as a token in itself. This means that the translation model treats, say, the Hindi word *pustak* (book) completely independent of the word *pustakein* (books). Any instance of *pustak* in the training data does not add any knowledge to the translation of *pustakein*. In the extreme case, while the translation of *pustak* may be known to the model, the word *pustakein* may be unknown and the system will not be able to translate it. While this problem does not show up as strongly in English due to the very limited morphological inflection in English, it does constitute a significant problem for morphologically rich languages such as Hindi, Telugu, Tamil etc. Lemmatization can reduce data sparseness, and allow inflectional variants of the same word to explicitly share a representation in the model. In principle, neural models can learn that inflectional variants are semantically related, and represent them as similar points in the continuous vector space [37]. However, while this has been demonstrated for high-frequency words, we expect that a lemmatized representation increases data efficiency. We verify the use of lemmas in both word based model and also in a subword model.

3.3.2 Morphological Features

Machine Translation suffers data sparseness problem when translating to/from morphologically rich and complex languages such as Hindi. Thus morphological analysis may help to handle data sparseness and improve translation quality. Different word types in Hindi have different sets of morph features. For example, verbs have person, number, gender, tense, aspect and nouns have case, number, gender. For some words the features may also be underspecified. Therefore, we treat the concatenation of all morphological features of the word as a string and treat this string as a separate feature value for each word along with other linguistic features.

3.3.3 POS Tags

Linguistic resources such as part-of-speech (POS) tags have been extensively used in statistical machine translation (SMT) frameworks and have yielded better performances. However, usage of such linguistic annotations in neural machine translation (NMT) systems has not been explored much. POS tags provide the linguistic knowledge and the syntactic role of each token in the context, which helps in information extraction and reducing data ambiguity.

3.3.4 Using Word-level Features in the Subword Model

Neural Machine Translation relies on first mapping each word into the vector space, and traditionally we have a word vector corresponding to each word in a fixed vocabulary. Addressing the problem of data scarcity and the hardness of the system to learn high quality

representations for rare words, [38] proposed to learn subword units and perform translation at a subword level. Subword segmentation of words is achieved using Byte Pair Encoding (BPE) which has been shown to work better than UNK replacement techniques. In this work, we also experiment with subword models. With the help of BPE, the vocabulary size is reduced drastically, thereby decreasing the OOV (out of vocabulary words) rate and we no longer need to prune the vocabularies. After the translation, we do an extra post processing step to convert the target language subword units back to normal words. We find this approach to be very helpful in handling rare word representations when translating from Hindi to English.

But we note that in BPE segmentation, some subword units are potentially ambiguous, and can either be a separate word, or a subword segment of a larger word. Also, text is represented as a sequence of subword units with no explicit word boundaries. Explicit word boundaries are potentially helpful to learn which symbols to attend to, and when to forget information in the Transformer layers. We use an annotation of subword structure similar to popular IOB format for chunking and named entity recognition, marking if a subword unit in the text forms the beginning (B), inside (I), or end (E) of a word. A separate tag (O) is used if a subword unit corresponds to the full word. To incorporate the word-level linguistic features in a subword model, we copy the word's feature values to all of its subword units.

3.4 Experimental Settings

3.4.1 Dataset

In our experiments, we use IIT-Bombay [39] Hindi-English parallel data. The training corpus consists of data from mixed domains. There are roughly 1.5M samples in the training data from diverse sources, while the development and test sets are from news domains.

Table 3.1: Statistics of our processed parallel data.

Dataset	Sentences	Tokens
IITB Train	1,528,631	21.5M / 20.3M
IITB Test	2,507	62.3k / 55.8k
IITB Dev	520	9.7k / 10.3k

3.4.2 Data Processing

We use Moses [40] toolkit for tokenization and cleaning the English side of the data. Hindi side of the data is first normalized with Indic NLP library[3] followed by tokenization with the

[3]https://anoopkunchukuttan.github.io/indic_nlp_library/

same library. As our preprocessing step, we remove all the sentences of length greater than 80 from our training corpus and lowercase the English side of the data. We use BPE segmentation with 32k merge operations. We use *Hindi Shallow Parser* [4] to extract all the linguistic features (i.e POS tags, morph features, lemma) and annotate Hindi text with the same. We also remove all punctuations from both Hindi and English to avoid any possible errors thrown by the shallow parser. All the linguistic features are joined with the original word or subword using the pipe ("|")*symbol*.

3.4.3 Training Details

For all of our experiments, we use OpenNMT-py [41] toolkit. We use Transformer model with 6 layers in both encoder and decoder each with 512 hidden units. The word embedding size is set to 512 with 8 heads. The training is done in batches of maximum 4096 tokens at a time with dropout set to 0.3. We use Adam optimizer to optimize model parameters. We validate the model every 5,000 steps via BLEU [22] and perplexity on the development set.

Table 3.2: Size of embedding layer of linguistic features, in system that includes all features, and contrastive experiments that add a single feature over the baseline. The embedding layer size of the word or subword feature is set to bring the total size to 512.

Features	**Embedding size**	
	all	single
subword tags	6	5
POS tags	10	10
Morph Features	20	20
Lemma	100	150
Word or subword	*	*

The embedding layer size of the all the linguistic features used varies, and is set to bring the total embedding layer size to 512 so as to ensure that the performance improvements are not simply due to an increase in the number of model parameters. All the features have different vocabulary sizes and after performing various experiments we found the optimum embedding size for each of the features listed in table 3.2, which is basically a hyperparameter in our setting.

3.5 Results and Discussion

We report the results of usage of linguistic features both in a normal word based model and also in a subword model. We also perform contrastive experiments in which only a single feature

[4] *http://ltrc.iiit.ac.in/showfile.php?filename=downloads /shallow_parser.php*

is added to the baseline. Table 3.3 shows our main results for Hindi-English word based model. All the linguistic features added in isolation proved to be effective in improving the translation performance of the word based model. But the combination of all linguistic features together gave us the lowest improvement of 0.19 BLEU. Experiments demonstrates that the gain from the different features is not fully cumulative and the information encoded in different features overlaps.

Table 3.3: Contrastive experiments for a word based Hindi-English Transformer model with individual linguistic features.

System	**BLEU**
Word baseline	17.13
+POS tags	17.51 (+0.38)
+Lemma	**17.65 (+0.52)**
+Morph features	17.44 (+0.31)
+All features	17.32 (+0.19)

Table 3.4 shows our results for a subword Hindi-English model. Except the lemma feature, all other features used independently in the subword model shows significant BLEU improvements. The reason behind the lemma feature not being helpful in improving the translation performance can be the nature of subword model itself. Translation at subword level inherently captures the linguistic information at the root level. The best performance is achieved when using IOB tags, POS tags and morph features in a subword model.

Table 3.4: Contrastive experiments for a subword Hindi-English Transformer model with individual linguistic features.

System	**BLEU**
Subword baseline	18.47
+IOB tags	18.64 (+0.17)
+POS tags	19.11 (+0.64)
+Lemma	17.99 (-0.48)
+Morph features	19.02 (+0.55)
+IOB, POS tags and Morph features	**19.21 (+0.74)**
+All features	18.34 (-0.13)

We also conducted an ablation study (shown in table 3.5) to know the effect of linguistic features on the translation model with varying training data size. Our hypothesis was that the linguistic features will be much more helpful when operating on less data. But the results indicate that the gains from the addition of linguistic features is limited.

Table 3.5: Ablation study showing the effect of usage of linguistic features in a subword model with varying data size.

System	Data size=25k	Data size=150k	Data size=350k	Data size=750k
Word baseline	0.35	2.72	9.64	12.71
Subword baseline	0.43	3.50	10.37	13.62
+IOB tags	0.33	3.53	10.87	13.89
+POS tags	0.59	3.85	10.93	14.05
+Lemma	0.50	3.46	10.67	13.71
+Morph features	0.42	3.68	10.50	13.97
+IOB, POS tags and Morph features	0.74	3.64	10.61	14.09
+All features	0.71	4.07	11.22	13.31

We observe on the manual inspection of the translation samples that there is a significant improvement in the translation quality after using linguistic features with the baseline. In table 3.6, we show and compare the outputs generated by our baseline model with the subword model using linguistic features. In all these examples, we show that the linguistically informed translation model generates more fluent and meaningful translations than the baseline model. For example, for the Hindi input sentence "बल्कि कुत्ते को देखा जाना ही अविश्वसनीय है", the baseline model produced a translation as "but the dog is unbelievable"" which does not convey the correct meaning of the input Hindi sentence and is therefore not an acceptable translation. Whereas, after using the linguistic features, the model generated much more fluent and meaningful translation: "but it is unbelievable to see the dog" that also resembles with the reference translation "the fact that the dog was spotted is unbelievable" in the terms of meaning.

Table 3.6: Translation examples illustrating the effect of adding linguistic input features.

system	sentence
source	एमआरएफ लगातार दसवीं बार जेडी पावर पुरस्कार से सम्मानित
transliteration	emaareph lagaataar dasaveen baar jedee paavar puraskaar se sammaanit
reference	mrf has been awarded the jd power award for the tenth time in a row
subword baseline	mrf has been awarded for 10th consecutive term gd power award
+IOB tags, POS tags and morph features	mrf has been awarded with jd power award for the tenth consecutive term
source	यह समारोह तीन हफ्ते पहले ही हुआ है
transliteration	yah samaaroh teen haphte pahale hee hua hai
reference	this ceremony took place three weeks ago
subword baseline	this ceremony has been held three weeks ago
+IOB tags, POS tags and morph features	this event took place three weeks ago
source	बल्कि कुत्ते को देखा जाना ही अविश्वसनीय है
transliteration	balki kutte ko dekha jaana hee avishvasaneey hai
reference	the fact that the dog was spotted is unbelievable
subword baseline	but the dog is unbelievable
+IOB tags, POS tags and morph features	but it is unbelievable to see the dog
source	ब्लेयर का कहना है कि वे ब्रिटेन वापसी चाहते हैं
transliteration	bleyar ka kahana hai ki ve briten vaapasee chaahate hain
reference	blair says he would like uk comeback
subword baseline	they want to return to britain says blair
+IOB tags, POS tags and morph features	blair says they want to return to britain

3.6 Related Work

Factored translation models are often used in phrase-based SMT [42] as a means to incorporate extra linguistic information. However, neural MT can provide a much more flexible mechanism for adding such information. Because phrase-based models cannot easily generalize to new feature combinations, the individual models either treat each feature combination as an atomic unit, resulting in data sparsity, or assume independence between features, for instance by having separate language models for words and POS tags. In contrast, we exploit the strong generalization ability of neural networks, and expect that even new feature combinations, e.g. a word that appears in a novel syntactic function, are handled gracefully. Linguistic features have also been used in Neural MT but all of them have shown the effectiveness of usage of linguistic features with the RNN model [33, 34, 35]. However, it is essential to verify whether the strong learning capability of the current state-of-the-art Transformer models make the explicit linguistic features redundant or if they can be easily incorporated to provide further improvements in performance. Also, the effectiveness of linguistic features in building NMT models for a low resource language pair like Hindi-English where Hindi is a Morphologically rich language has not been shown earlier.

3.7 Summary

In this chapter, we investigated whether the linguistic input features are helpful in improving the translation performance of the state-of-the-art Transformer based NMT model, and our empirical results show that this is the case . We show our results on Hindi-English, a low resource language pair where Hindi is a morphologically rich and a free word order language whereas on the other end we have English which is morphologically less complicated and word order specific language. We empirically test the inclusion of various linguistic features, including lemmas, part-of-speech tags, morphological features and IOB tags for a (sub)word model. Our experiments show that linguistic input features yield significant improvements of +0.74 over a subword baseline and +0.52 over a word based NMT baseline.

In the future, we expect several developments on the usefulness of linguistic input features in neural machine translation. The machine learning capability of neural architectures is likely to increase, decreasing the benefit provided by the features we tested. Therefore in future, we will need better methods to incorporate linguistic information in such neural architectures. In the next chapter, we discuss the problem of data scarcity in English to Indian Language NMT and propose methods to alleviate the problem.

Chapter 4

Efficient Neural Machine Translation for Low Resource Languages via Exploiting Related Languages

Neural Machine Translation (NMT) is a rapidly advancing MT paradigm and has shown promising results for many language pairs, especially in large training data scenarios. Since, the condition of large parallel corpora is not met for Indian-English language pairs, we present our efforts towards building efficient Neural Machine Translation systems between Indian languages (specifically Indo-Aryan languages) and English via efficiently exploiting parallel data from the related languages. We propose a technique called Unified Transliteration and Subword Segmentation to leverage language similarity while using parallel data from related language pairs. We also propose a Multilingual Transfer Learning technique to leverage parallel data from multiple related languages to assist translation for low resource language pair of interest. Our experiments demonstrate an overall average improvement of 5 BLEU points over the standard Transformer based NMT baseline.

4.1 Introduction

In recent years, Neural Machine Translation [30, 19, 31, 32, 1] (NMT) has become the most prominent approach to Machine Translation (MT) due to its simplicity, generality and effectiveness. In NMT, a single neural network often consisting of an encoder and a decoder is used to directly maximize the conditional probabilities of target sentences given the source sentences in an end-to-end paradigm. NMT models have been shown to surpass the performance of previously dominant statistical machine translation (SMT) [12] on many well-established translation tasks.

However, in order to reach high accuracies, neural translation systems tend to require very large parallel training corpora [21]. As a matter of fact, such corpora are not yet available for many language pairs. Indian languages are not an exception to this, however they are extremely diverse, belonging to different language families, employing various scripts and spanning across

a multitude of dialects. They are broadly classfied into two language families: Indo-Aryan (e.g. Hindi, Gujarati, Bengali, Punjabi and Marathi) and Dravidian (e.g. Telugu, Tamil, Kannada and Malayalam). The majority of Indian languages are morphologically rich and depict unique characteristics, which are significantly different from languages such as English. Some of these characteristics are the relatively free word-order with a tendency towards the Subject-Object-Verb (SOV) construction and a high degree of inflection. Also when translating between morphologically rich & free word order languages like Indian languages and the other end of morphologically less complicated & word order specific languages like English, the well-known issues of missing words and data sparsity arise; and hence affect the accuracy of translation and leads to more out-of-vocabulary (OOV) words.

Since, NMT models learns poorly from low resource corpora, building effective NMT systems for low resource languages (e.g. Indian languages) becomes a primary challenge. The bulk of research on low-resource NMT has focused on exploiting monolingual data, or parallel data involving other language pairs. Some of the most famous methods to improve NMT models with monolingual data range from backtranslation [43], dual learning [44] to Unsupervised MT [45, 46, 47]. Similarly, parallel data from other languages can be used to either pretrain the network or jointly learn the representations [2, 48, 49, 31].

Currently, transfer learning (TL) is being widely used for low resource language translation because it is one of the vital directions for addressing data sparsity problem in low resource NMT [2, 50, 51, 48]. However, the most of the existing approaches that take advantage of transfer learning have a major limitation: they do not exploit multiple languages together and efficiently. The idea presented by [2] may have the shortcoming of exploiting only one high-resource model (parent) at a time to optimize the low resource model (child). Actually, the use of highly related multiple language pairs might help in increasing the translation quality of the child model. Also, the original Transfer Learning method [2] makes no assumption about the relatedness of the parent and child languages. Multilingual Neural Machine Translation [49, 31] approaches which also use parallel data from many different languages to improve the translation quality of NMT models do not exploit kind of language similarity.

In this work, we present our efforts towards building efficient Neural Machine Translation systems between Indian languages (specifically Indo-Aryan languages) and English by exploiting parallel data from related languages. We aim to deal with the problem of how to make full use of these corpora of highly related languages, to increase the translation quality of low resource languages. Towards this end, we introduce two simple and yet effective approaches:

1). Multilingual Transfer Learning: to enable the low resource languages (child model) to exploit parallel data from multiple related languages which may or may not be high resourced, and

2). Unified Transliteration and Subword Segmentation: to exploit language similarity between the related languages.

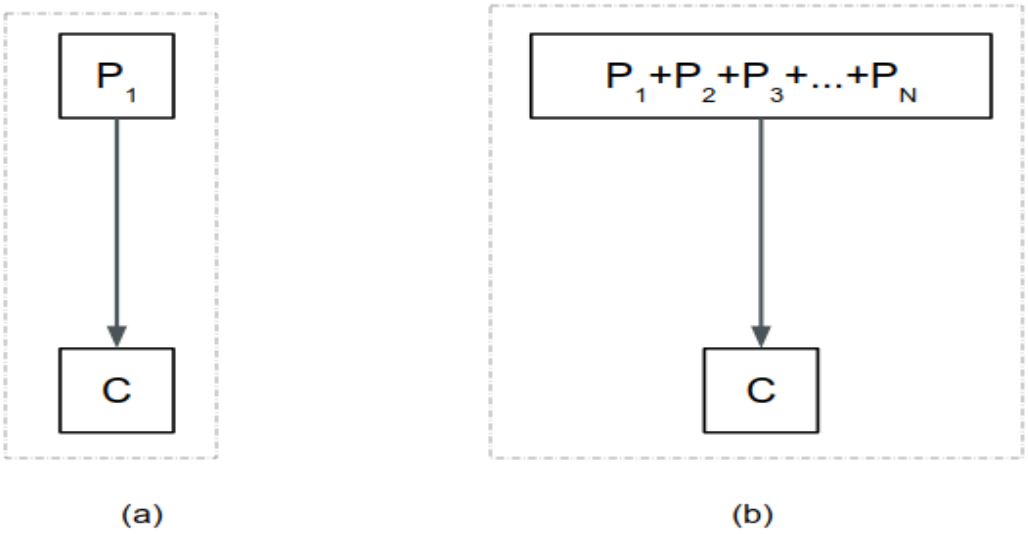

Figure 4.1: Model architecture of Multilingual Transfer Learning approach. Single initialized NMT (Figure 1(a)) fine-tunes the child model only with a single high-resource parent model [2]. In the main architecture (Figure 1(b)) of our proposed approach, Multilingual Transfer Learning, a multilingual NMT model is first constructed with various language pairs that are semantically, syntactically similar, and share many words with the child model. This multilingual parent model, which may or may not contain the child language pair, is then finetuned on the low resource language pair of interest (i.e. child model).

Experiments show that our approaches are effective for English to Indian language translation (for both the translation directions) and significantly outperform the state-of-the-art Transformer [1] baseline by almost 5 BLEU points. Also, our proposed approach of Multilingual Transfer Learning significantly outperforms the simple Transfer Learning [2] approach.

4.2 Methodology

The core idea of our method is to extend the Multilingual learning [31] and Transfer Learning [2] approaches to effectively exploit parallel data from multiple related languages. In section 4.3.2, we explain our Unified Transliteration and Subword Segmentation technique to exploit language relatedness among the parallel data of related languages. Section 4.3.3 and 4.3.4 describe our modified Multilingual Learning and Transfer Learning techniques for NMT. In section 4.3.5, we describe our Multilingual Transfer Learning approach.

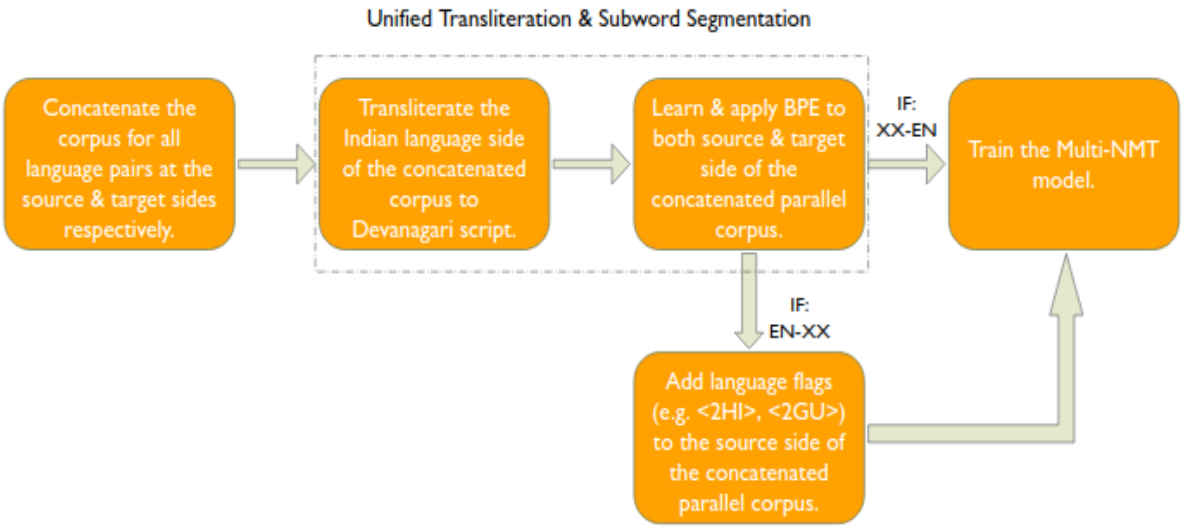

Figure 4.2: Our pipeline for building Multilingual NMT models for Indian languages.

4.2.1 Language Relatedness

In this work, we experiment on Indo-Aryan languages specifically Hindi, Punjabi, Gujarati, Marathi & Bengali. Being from one language family, these languages are very much related to each other and share many features. These languages are morphologically rich and depict unique characteristics, which are significantly different from languages such as English. Some of these characteristics are the relatively free word-order with a tendency towards the Subject-Object-Verb (SOV) construction, a high degree of inflection, usage of reduplication and conjunct verbs. These languages share many common words which have same root and meaning. They use different Indic scripts derived from the ancient Brahmi script, but correspondences can be established between equivalent characters across scripts.

4.2.2 Unified Transliteration and Subword Segmentation

Unlike the original Transfer Learning [2] and the Multilingual Neural MT [31] methods which do not exploit any language relatedness, the basic idea of this approach is to exploit the relationship between the related language lexicons while using parallel data from related languages to assist translation of low resource languages. To do so, we find a representation of the data that ensures a sufficient overlap between the vocabularies of the related languages.

Since, the languages involved in the models have different orthographies, the data processing should help to map them into a common orthography. But here we take a minimalist approach. We transliterate all the Indian languages, that is Hindi, Gujarati, Bengali, Marathi & Punjabi into a common Devanagari script to share the same surface form. This unified transliteration is

a string homomorphism, replacing characters in the all languages mentioned above with Hindi characters (script conversion to Devanagari) or consonant clusters independent of context.

Now, to increase the overlap between the vocabularies of the languages used in a model, which are already transliterated into a common script and consequently share the same surface form, we use Byte Pair Encoding (BPE) [38] to break words into subwords. For the BPE merge rules to not only find the common subwords between two related languages but also ensure consistency between source and target segmentation among each language pair, we learn the rules from the union of source and target data of all the language pairs involved in the model construction. The rules are then used to segment the corpora. It is important to note that this results in a single vocabulary, used for both the source and target languages in all the language pairs.

4.2.3 Multilingual Learning for NMT

The objective of Multilingual Learning for NMT is to construct a single model for translating to and from multiple languages. Early works in multilingual NMT utilizes separate encoder, decoder and an attention mechanism to support the translation of either one-to-many or many-to-one language directions. [49] introduced a many-to-many system, however, relying upon separate encoder-decoder setup with a shared attention mechanism. In a simplified manner & yet delivering better performance, [31] introduced a "language flag" based approach that shares the attention mechanism and a single encoder-decoder network to enable multilingual models. A language flag or token is prepended to the input sequence to indicate which direction to translate to. The decoder learns to generate the target given this input.

But these Multilingual NMT approaches do not explicitly exploit any kind of language similarity or how many shared words there are among the different source and target languages. Mainly, they aim on exploiting many different source and target languages rather than focusing on similarities between many languages that are used in the training and the languages that is used in testing. So, we modify the Multilingual NMT approach [31] with Unified Transliteration and Subword segmentation technique to exploit the language similarity between the Indo-Aryan languages. We experiment with this modified approach (as described in Figure 4.2) in our work on efficient NMT for Indian languages.

4.2.4 Transfer Learning for NMT

[2] proposed how Transfer Learning between two NMT models can improve a low-resource NMT task. In their approach, a language pair with the relatively large amount of parallel data is first utilized to train a parent model and we call this phase as "pretraining". Then the encoder-decoder parameters are transferred to initialize a child model for a low-resource language pair of interest. After initializing, the model enters the "finetuning" stage, where child model is

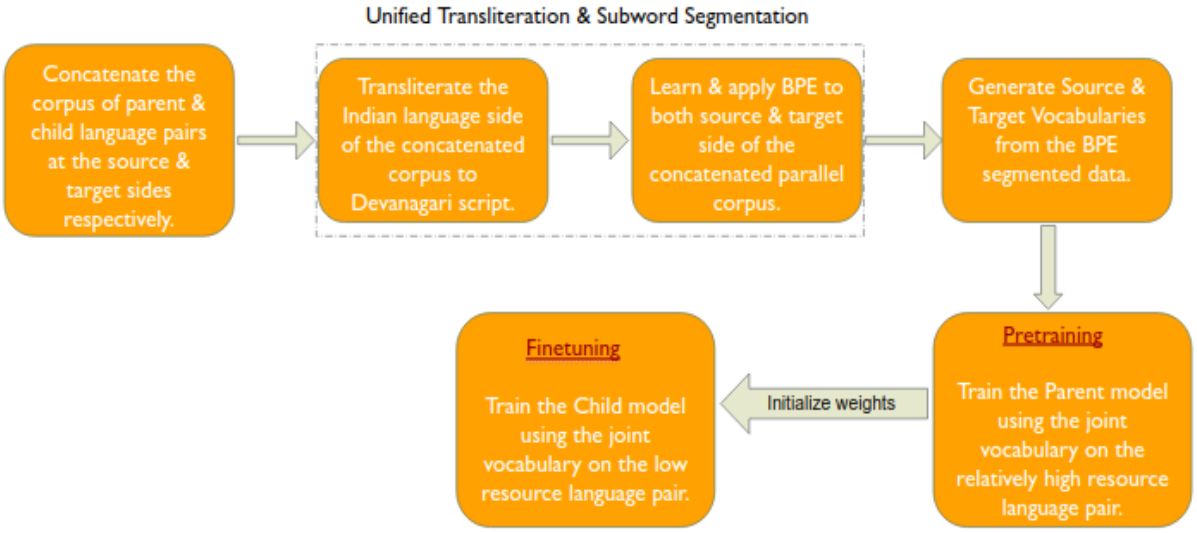

Figure 4.3: Our pipeline for building Transfer Learning models for Indian languages.

finetuned on low resource language pair. This enables the inductive transfer of knowledge from the parent model to the child model. This approach do not make any assumption between the relatedness of parent and child language pair. However, in our work, we use a relatively high resource language pair as parent model which has similar syntactical and morphological properties as the child language pair. We further exploit the language relatedness of parent and child language pairs via using Unified Transliteration and Subword Segmentation technique for better knowledge transfer. We experiment with this modified Transfer Learning technique (as described in Figure 4.3) and demonstrate huge BLEU improvements for low resource Indian languages.

4.2.5 Multilingual Transfer Learning for NMT

In a normal Transfer Learning [2] approach for NMT, the parent model is trained on a single high resource language pair which may or may not be related to the child language pair of interest. [50] presented a double transfer learning technique which first trains a parent model on a single high resource language pair, then initializes the next parent model on the same single high resource language pair but with different domain and corpus size. But none of these Transfer Learning approaches exploit parallel data from multiple languages. However, learning from multiple languages can result in better knowledge transfer.

Therefore, in this work, we propose a new Transfer Learning approach called as Multilingual Transfer Learning (refer to Figure 4.1) to enable the low resource languages to efficiently learn from multiple related languages which may or may not be high resourced. In this approach,

the parent model is a Multilingual NMT model of related languages and also the child language pair. This Multilingual parent NMT model also uses the Unified Transliteration and Subword Segmentation technique to exploit language relatedness more efficiently as discussed in section 4.3.3. After pretraining the multilingual parent model, the child model is initialzed with parent model parameters and is then finetuned on the low resource language pair of interest.

The proposed approach may deliver better results than Multilingual NMT and Transfer Learning because adding more languages into one model may result in better knowledge transfer (i.e multilingual NMT) but it can also result in ambiguities between languages at the inference time. So a multilingual NMT model finetuned on the language pair of interest can potentially remove all the inconsistencies at the inference time.

4.3 Experimental Settings

4.3.1 Dataset

In our experiments, we use IIT-Bombay [39] parallel data for Hindi-English. The training corpus consists of data from mixed domains. We use multilingual ILCI (Indian Language Corpora Initiative) corpus [52], which contains roughly 50,000 parallel sentences for each of the Indian language i.e. Gujarati, Punjabi, Marathi, Bengali and also for English. The ILCI data is from tourism and health domains. For every XX-EN language pair (where XX is: Gujarati, Marathi, Bengali or Punjabi), the English side of the data is same because of the multilingual nature of the corpus. We check & clean the ILCI corpus manually as it contains a lot of misalignments and mistranslations.

Table 4.1: Statistics of our cleaned & processed parallel data, where XX is: Gujarati, Marathi, Bengali or Punjabi

Dataset	**Sentences**
IITB HI-EN Train	1,528,631
ILCI XX-EN Train	46,490
ILCI XX-EN Test	2,000
ILCI XX-EN Dev	500

4.3.2 Data Processing

We use Moses [40] toolkit for tokenization and cleaning the English side of the data. All the Indian language data is first normalized with Indic NLP library[1] followed by tokenization with the same library. As our preprocessing step, we remove all the sentences of length greater than

[1]https://anoopkunchukuttan.github.io/indic_nlp_library/

Table 4.2: BLEU scores of the contrastive experiments for Indian Language to English translation (XX to EN).

No.	Model Description	Gujarati	Bengali	Marathi	Punjabi
1	Baseline	28.37	22.40	25.29	30.51
2	Multilingual Model of all ILCI data	25.14	21.47	23.56	25.43
3	Multilingual Model of IITB HI-EN data & all ILCI data	28.62	22.71	26.90	29.46
4	Multilingual Model of IITB HI-EN data & ILCI data of XX-EN	29.18	23.93	27.15	30.54
5	Finetuning 2nd model on XX-EN	26.83	22.72	25.36	27.12
6	Finetuning 3rd model on XX-EN	**33.78 (+5.41)**	**27.55 (+5.15)**	**31.79 (+6.5)**	**34.70 (+4.19)**
7	Finetuning 4th model on XX-EN	33.72	27.40	31.80	34.68
8	Finetuning model pretrained on IITB HI-EN data on XX-EN	33.13	27.06	31.27	34.54

80 from our training corpus & lowercase the English side of the data. In all cases, we use BPE segmentation with 16k merge operations as described in section 3.3.2.

4.3.3 Training Details

For all of our experiments, we use OpenNMT-py [41] toolkit. We use Transformer model with 6 layers in both encoder and decoder each with 512 hidden units. The word embedding size is set to 512 with 8 heads. The training is done in batches of maximum 4096 tokens at a time with dropout set to 0.3. We use Adam optimizer to optimize model parameters. We validate the model every 5,000 steps via BLEU [22] and perplexity on the development set. We train all our NMT models for 150k steps except for fintuning which is done for 10k steps. After translation at the test time, we rejoin the translated BPE segments and converted the translated sentences back to their original language scripts. Finally, we evaluated using BLEU score.

4.4 Results and Discussion

We report the results of our experiments using Multilingual Learning, Transfer Learning and Multilingual Transfer Learning techniques on Gujarati-English, Bengali-English, Marathi-English and Punjabi-English language pairs for both the translation directions (XX-EN and EN-XX). Table 3.2 shows our main results for Indian language to English (XX-EN) translation direction. Multilingual models for XX-EN language direction doesn't show any improvements over the baseline. The reason can be the multilingual nature of the ILCI data, where for each English sentence on the target side, there are 4 different sentences on the source side, each in different language, thereby creating ambiguities in the model. The transfer learning model built using Unified Transliteration and Subword Segmentation that was trained on IITB HI-EN data and then finetuned on XX-EN data (see model no. 8 in table 4.2) results in an average improvement of 5 BLEU scores over the simple NMT baseline.

Table 4.3: BLEU scores of the contrastive experiments for English to Indian Language translation (EN to XX).

No.	Model Description	Gujarati	Bengali	Marathi	Punjabi
1	Baseline	20.67	16.59	15.13	25.20
2	Multilingual Model of all ILCI data	24.61	19.81	17.92	28.02
3	Multilingual Model of IITB EN-HI data & all ILCI data	20.63	16.51	15.05	21.76
4	Multilingual Model of IITB EN-HI data & ILCI data of EN-XX	14.30	6.38	8.88	14.54
5	Finetuning 2nd model on EN-XX	24.75	20.25	18.75	28.16
6	Finetuning 3rd model on EN-XX	**26.22 (+5.55)**	**21.62 (+5.03)**	**19.90 (+4.77)**	**30.27 (+5.07)**
7	Finetuning 4th model on EN-XX	25.52	20.45	19.77	29.53
8	Finetuning model pretrained on IITB EN-HI data on EN-XX	25.35	21.77	19.58	29.54

Table 4.3 shows our main results for English to Indian language (EN-XX) translation direction. In this case, multilingual model of all ILCI data shows significant improvements over the baseline, unlike in XX-EN translation direction. The reason is that in EN-XX direction, language tokens are used on the source side which guides the decoder to which direction should the model translate to whereas the same is not possible for XX-EN direction as verified by our preliminary experiments. The other two multilingual models containing IITB EN-HI data, show performance degradation, potentially due to the mismatch between the size of the IITB EN-HI (1.5M sentences) & ILCI data (47k sentences). The transfer learning model that was trained on IITB EN-HI data and then finetuned on EN-XX data (see model no. 8 in table 4.3) also resulted in an average improvement of 5 BLEU scores.

In both the translation directions, the transfer learning models prove to be very effective in improving the translation quality by almost 5 BLEU points. In most cases, the multilingual models doesn't prove to be effective. However, as discussed in section 4.3.5, finetuning the multilingual models (i.e. Multilingual Transfer Learning) on XX-EN or EN-XX data, removes some potential ambiguities in the model and shows significant improvements than their simple multilingual model counterparts. The best performance (almost 5-6 BLEU improvements over the baseline) is achieved by finetuning the multilingual model (trained on IITB HI-EN or EN-HI data and all the ILCI data) on EN-XX or XX-EN outperforming all the simple NMT, Multilingual NMT and Transfer Learning baselines.

We observe on the manual inspection of the translation samples that there is a significant improvement in the translation quality than the baseline sytem after finetuning the Multilingual model trained on IITB Hi-En data and all the ILCI data (i.e Multilingual Transfer Learning). In tables 4.4, 4.5, 4.6 and 4.7, we show and compare the translation outputs generated by our baseline model with the outputs generated by our Multilingual Transfer Learning models for all the languages that we experimented on. In all these examples, we show that the Multilingual Transfer Learning model generates more fluent and meaningful translations than the baseline model. For example, for the Punjabi sentence "ਪ੍ਰਿਥਵੀ ਦਾ ਸਭ ਤੋਂ ਉੱਚਾ ਪ੍ਰਾਣੀ ਜ਼ਿਰਾਫ ਨੂੰ ਦੇਖਣ ਦੀ ਤਮੰਨਾ ਪੂਰੀ ਨਹੀਂ ਹੋਈ ਸੀ ।"" shown in table 4.4, the baseline model outputs the translation as

“The highest zoological garden of the earth was not complete to see giraffe.” which carries wrong information such as words like “zoological garden” and does not convey the right meaning of the source sentence. But the translation from the Multilingual Transfer Learning model “The wish to see giraffe the highest animal of earth was not fulfilled.” seems appropriate and is almost same as the reference translation “Our desire to see the tallest animal on earth giraffe had not been fulfilled yet.”

Table 4.4: Punjabi-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system.

system	sentence
source	ਵਹੇਲਾਂ ਦੀ ਸਮਝਦਾਰੀ, ਉਹਨਾਂ ਦੀਆਂ ਅਵਾਜ਼ਾਂ, ਉਹਨਾਂ ਦੇ ਭੋਜਨ, ਵਿਵਹਾਰ ਆਦਿ ਨੂੰ ਲੈਕੇ ਕਈ ਖੋਜ ਦੁਨੀਆਭਰ ਵਿਚ ਹੁੰਦੇ ਆਏ ਹਨ ।
transliteration	Vahēlāṁ dī samajhadārī, uhanāṁ dī'āṁ avāzāṁ, uhanāṁ dē bhōjana, vivahāra ādi nū laikē ka'ī khōja dunī'ābhara vica hudē ā'ē hana.
reference	Several researches have been conducted about the intelligence of whales, their voice, their food, behavior etc.
baseline	Several research of the youth, their voices, their food, behavior etc. have come to the world over.
Multilingual Transfer Learning	Many discoveries have been happening worldwide regarding the wisdom of whale, their voices, their food, behavior etc.
source	ਪਹਿਲਾਂ ਇਹ ਗੱਡੀਆਂ ਸਿਤੰਬਰ ਤੋਂ ਅਪਰੈਲ ਤੱਕ ਚੱਲਦੀਆਂ ਸੀ ।
transliteration	Pahilāṁ iha gaḍī'āṁ sitabara tōṁ aparaila taka caladī'āṁ sī.
reference	Initially these trains used to run from September to April.
baseline	First this cars run from September to April.
Multilingual Transfer Learning	Initially these cars used to run from September to April.
source	ਪ੍ਰਿਥਵੀ ਦਾ ਸਭ ਤੋਂ ਉੱਚਾ ਪ੍ਰਾਣੀ ਜ਼ਿਰਾਫ ਨੂੰ ਦੇਖਣ ਦੀ ਤਮੰਨਾ ਪੂਰੀ ਨਹੀਂ ਹੋਈ ਸੀ ।
transliteration	Prithavī dā sabha tōṁ ucā prāṇī zirāpha nū dēkhaṇa dī tamanā pūrī nahīṁ hō'ī sī.
reference	Our desire to see the tallest animal on earth giraffe had not been fulfilled yet.
baseline	The highest zoological garden of the earth was not complete to see giraffe.
Multilingual Transfer Learning	The wish to see giraffe the highest animal of earth was not fulfilled.

Table 4.5: Bengali-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system.

system	sentence
source	খাবার খুব চিবিয়ে খান এবং তা ভালো করে হজম হতে দিন।
transliteration	Khābāra khuba cibiẏē khāna ēbaṁ tā bhālō karē hajama hatē dina
reference	Eat food chewing well and allow it to digest well.
baseline	Eat food very well and eat that digests properly.
Multilingual Transfer Learning	Eat food chewing well and let it digest properly.
source	পরিসরে বোটিংএর সুবিধাও আছে।
transliteration	Parisarē bhōṭiṁ ēra subidhā āchē.
reference	Facility of boating is also there in the complex.
baseline	There is the facility of boating also in the temple.
Multilingual Transfer Learning	The facility of boating is also available in the complex.
source	এর থেকে রক্ষা পাওয়ার জন্য আবহাওয়া পরিবর্তণ হওয়া মাত্রই বডি লোশন ব্যবহার করা শুরু করে দেওয়া উচিত।
transliteration	Ēra thēkē rakṣā pā'ōẏāra jan'ya ābahā'ōẏā paribartaṇa ha'ōẏā mātra'ī baḍi lōśana byabahāra karā śuru karē dē'ōẏā ucita.
reference	To prevent it one should start using winter body lotion as soon as the season changes.
baseline	In order to evade this the weather began to use the body lotion soon.
Multilingual Transfer Learning	To prevent it one should start using body lotion just as the weather change.

4.5 Summary

In this chapter, we explored effective methods to exploit parallel data from multiple related languages to improve the translation between Indian languages and English. Our results show that Multilingual Learning for translation between Indian Languages and English is not much effective, maybe due to the nature of the data we have. But the performance of multilingual models can be easily enhanced by finetuning them on the low resource language pairs of interest.

Our experiments show that using a Multilingual NMT model as a parent model (consisting of multiple language pairs with related languages either on the source side or on the target side) and finetuning it on the low resource language pair of interest yields an overall average improvement of 5 BLEU points over a standard Transformer base NMT baseline. Our proposed Multilingual Transfer Learning approach also outperforms the Transfer Learning approach by significant BLEU points. In future, we would like to work on effective techniques to exploit monolingual data and parallel data from other languages together to improve the translation of low resource languages.

Table 4.6: Gujarati-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system.

system	**sentence**
source	િવનેરીના કિલ્લામાં છત્રપતિ શિવાજીનો જન્મ થયો હતો .
transliteration	Ivanērīnā killāmāṁ chatrapati śivājīnō janma thayō hatō.
reference	Chhatrapati Shivaji was born in the fort of Shivneri.
baseline	In the fort of Shivji there was birth of Chhatrapati Shivaji.
Multilingual Transfer Learning	Chhatrapati Shivaji was born in the fort of Shivanri.
source	લોહીદબાણ સામાન્ય કરતાં ઓછું થઈ જાય છે .
transliteration	Lōhīdabāṇa sāmān'ya karatāṁ ōchuṁ tha'ī jāya chē.
reference	Blood pressure becomes lesser than normal.
baseline	Blood pressure reduces on normal.
Multilingual Transfer Learning	Blood pressure becomes less than normal.
source	મળ અપાન વાયુ સાથે પણ નીકળી જાય છે .
transliteration	Maḷa apāna vāyu sāthē paṇa nīkaḷī jāya chē.
reference	Stool comes out with flatus also.
baseline	Excretion also comes out with air.
Multilingual Transfer Learning	Stool comes out with air also.

Table 4.7: Marathi-English Translation examples illustrating the quality difference between the translation generated by Multilingual Transfer Learning and the baseline NMT system.

system	sentence
source	अमृतसरमध्ये तुमचा खिसा आणि इच्छेनुसार प्रत्येक प्रकारचे विश्रांतीगृह उपलब्ध आहेत .
transliteration	Amrtasaramadhyē tumacā khisā āṇi icchēnusāra pratyēka prakāracē viśrāntīgrha upalabdha āhēta.
reference	In Amritsar there are all types of hotels present according to your pocket and desire.
baseline	According to your pocket and wish entire hotels are available in Amritsar.
Multilingual Transfer Learning	In Amritsar as per your pocket and wish all kinds of hotels are available.
source	केवळ पावसाळ्याच्या ऋतुत पुर आणि सापांचा दंश मारण्याचा धोका जास्त असतो .
transliteration	Kēvaḷa pāvasāḷyācyā rtuta pura āṇi sāpāñcā danśa māraṇyācā dhōkā jāsta asatō.
reference	Only in the season of monsoons there remains a danger of flood or snake biting.
baseline	Not only in the season of monsoon the danger of spotting and bite is more.
Multilingual Transfer Learning	In the rainy season only the danger of flood and snake bites is high.
source	हीच अवशेष ऊर्जा शरीराच्या वसायलांमध्ये जमा होऊन स्थूलपणा वाढवते .
transliteration	Hīca avaśēṣa ūrjā śarīrācyā vasāyalāmmadhyē jamā hō'ūna sthūlapaṇā vāḍhavatē.
reference	This very remaining energy increases obesity accumulating in the fatty places of body.
baseline	This very ruins increases the fat accumulation in the activity of the body.
Multilingual Transfer Learning	This residue energy increases obesity by accumulating in the tissues of the body.

Chapter 5

The IIIT-H Gujarati-English Machine Translation system for WMT19

In this chapter, we describe our work on building efficient NMT system for Gujarati-English language pair. Given the fact that the two languages belong to different language families and there is not enough parallel data for this language pair, building a high quality NMT system for this language pair is a difficult task. Towards this end, we leverage training data from Hindi-English language pair to assist translation for Gujarati to English via developing a Multilingual Translation model. Experiments reveal that our approach helps in significant BLEU improvements upto 11.5 over the baseline NMT.

5.1 Introduction

Neural Machine Translation [30, 19, 31, 32, 1] has been receiving considerable attention in the recent years, given its superior performance without the demand of heavily hand crafted engineering efforts. NMT often outperforms Statistical Machine Translation (SMT) techniques but it still struggles if the parallel data is insufficient like in the case of Indian languages.

The bulk of research on low resource NMT has focused on exploiting monolingual data or parallel data from other language pairs. Some recent methods to improve NMT models that exploit monolingual data ranges from back-translation [53], dual NMT [44] to Unsupervised MT models [45, 46, 47]. Transfer Learning is also a promising approach for low resource NMT which exploits parallel data from other language pairs [2, 51, 48]. Typically it is achieved by training a parent model in a high resource language pair, then using some of the trained weights as the initialization for a child model and further train it on the low-resource language pair. Other promising approach for improving translation performance for low resource languages is Multilingual Neural Machine Translation. It has been shown that exploiting data from other language pairs & joint training helps in improving the translation performance of NMT models. [54, 55, 31].

This chapter describes the NMT system of our team (IIIT-H) for WMT19 Gujarati→English news translation task. We used both an attention-based encoder-decoder model and transformer model as our baseline systems and used Byte Pair Encoding (BPE) to enable open vocabulary translation. We then leverage Hindi-English parallel corpus in a multilingual setting so as to improve our baseline system. We basically combined Hindi-English and Gujarati-English parallel corpus and use it as our training corpus. Our multilingual system is similar to [31] but we don't use any artificial token at the start of source sentences to indicate the target language. The reason is trivial, that is we have only English as our target language.

5.2 Multilingual Neural Machine Translation

Most of the practical applications in Machine Translation have focused on individual language pairs because it was simply too difficult to build a single system that translates to and from many language pairs. But Neural Machine Translation was shown to be an end-to-end learning approach and was quickly extended to multilingual machine translation in several ways. In [56], the authors modify the attention-based encoder-decoder approach by introducing separate decoder and attention mechanism for each target language. In [57], multi-source translation was proposed where the model has different encoders and different attention mechanisms for different source languages. In [55], the authors proposed a multi-way multilingual NMT model using a single shared attention mechanism but with multiple encoders/decoders for each source/target language. In this work, we adopted the approach proposed in [31], where a single NMT model is used for multilingual machine translation. We used Hindi-English as our assisting language pair and combined it with Gujarati-English parallel data to form a multi source translation system.

5.3 Experimental setup

5.3.1 Dataset

In our experiments, we use the Gujarati-English training data provided by the organisers namely Wiki Titles, Bible corpus, Localisation Opus, Wikipedia corpus & crawled corpus. It consists of around 155K parallel sentences. We used newsdev2019 as our development corpus. For building our multilingual model, we used IIT-Bombay parallel data [39] as our Hindi-English parallel corpus. The top level statistics of the data used is provided in Table 5.1.

Table 5.1: Statistics of our processed parallel data.

Dataset	Sentences	Tokens
IITB Hi-En Train	15,28,631	21.5M / 20.3M
Gu-En Train	1,55,767	1.68M / 1.58M
Gu-En Dev	1,997	51.3K / 47.4K
Gu-En Test	1,998	51.5K / 47.5K

5.3.2 Data Processing

We used Moses [40] toolkit for tokenization and cleaning the English side of the data. Gujarati and Hindi sides of the data is first normalized with Indic NLP library[1] followed by tokenization with the same library. As our pre-processing step, we removed all the sentences of length greater than 80 from our training corpus.

5.3.3 Subword Segmentation for NMT

Neural Machine Translation relies on first mapping each word into the vector space, and traditionally we have a word vector corresponding to each word in a fixed vocabulary. Addressing the problem of data scarcity and the hardness of the system to learn high quality representations for rare words, [38] proposed to learn subword units and perform translation at a subword level. With the goal of open vocabulary NMT, we incorporate this approach in our system as a preprocessing step. In our early experiments, we note that Byte Pair Encoding (BPE) works better than UNK replacement techniques. For our baseline system, we learn separate vocabularies for Hindi and English each with 32k merge operations. For our multilingual model, we learn a joint vocabulary for Hindi and Gujarati & a separate vocabulary for English. With the help of BPE, the vocabulary size is reduced drastically and we no longer need to prune the vocabularies. After the translation, we do an extra post processing step to convert the target language subword units back to normal words. We found this approach to be very helpful in handling rare word representations.

5.3.4 Script Conversion

India is a linguistically rich country having 22 constitutional languages, written in different scripts. Indian languages are highly inflectional with a rich morphology, default sentence structure as subject object verb (SOV) and relatively free word order. Many of them are structurally similar, also called as sibling languages. Hindi & Gujarati languages are such siblings. That is why, we have chosen Hindi as an assisting language for our multilingual model.

[1] https://anoopkunchukuttan.github.io/indic_nlp_library/

Although, there are many linguistic similarities between Gujarati & Hindi, both of these languages are written in different scripts. So, to make a strong multilingual NMT model, we converted the script of the Gujarati side of the parallel corpus to Hindi (Devanagari script). We used Indic NLP Library's transliteration script for this purpose. We found this approach to be very helpful in enabling better sharing between languages on the encoder side. BPE also enhances the usage of script conversion technique. We used script conversion only with our additional Multilingual NMT experiments based on Transformer architecture.

5.3.5 Training Details

The structure of our NMT model is same as in [30], an RNN based encoder-decoder model with Global Attention mechanism. We used an LSTM based Bi-directional encoder and a unidirectional decoder. We kept 4 layers in both the encoder & decoder with embedding size set to 512. The batch size was set to 64 and a dropout rate of 0.3. We used Adam optimizer [58] for our experiments. Our multilingual model is trained with all the same hyperparameters as our baseline model except that the training data is a combination of Hindi-English & Gujarati-English parallel data.

5.4 Results

In this section, we report the BLEU [22] scores on the test sets provided in WMT19. Our simple NMT model which is an attention-based LSTM encoder-decoder model achieves a BLEU score of 6.2 on the test set. Our multilingual model which is trained with the help of Hindi-English parallel corpus attains a BLEU score of 9.8, showing a gain of +3.6 BLEU points on the same test set.

Table 5.2: WMT19 evaluation of our systems

System	BLEU
encoder-decoder + attention	6.2
Multilingual model	**9.8(+3.6)**

5.5 Additional Transformer Experiments

In this section, we present a set of experiments and results post WMT19 shared task involving the Transformer [1] architecture. We used the Transformer-Base architecture in this set of experiments with the rest of the pipeline being kept same as described before. We used 6 layers in both the encoder decoder with embedding size set to 512. The batch size was 2048 tokens & a dropout of 0.3. We used Adam optimizer for our experiments. During inference time,

we averaged the checkpoints of the model at different epochs to obtain better results than a single checkpoint. In the multilingual Transformer experiments, we employ script conversion technique for its merits described before.

In table 5.3, we provide the results of our Transformer experiments and also compare it to other systems submitted to WMT19.

Table 5.3: Our Transformer models vs other systems at WMT19

System	**BLEU**
Transformer	4.28
Multilingual Transformer	15.78 (+11.5)
+ Averaging	**16.49 (+0.71)**
NICT (Unsupervised MT)	9.6
NICT (Transfer Learning)	18.6
NEU (WMT19 Best)	26.5

We observe on the manual inspection of the translation samples that there is a significant improvement in the translation quality after using the Multilingual model trained on IITB Hi-En data and the Gujarati-English data than the baseline system. In table 5.4, we show and compare the translation outputs generated by our baseline model with the outputs generated by our Multilingual models. In all these examples, we show that outputs generated by the baseline NMT model are very bad due to the low quantity of parallel data and utilizing Hi-En data along with Gujarati-English data generates much more fluent and meaningful translations than the baseline model.

Table 5.4: Gujarati-English Translation examples illustrating the quality difference between our baseline and Multilingual models.

system	**sentence**
source	જમ્મુ-કાશ્મીરમાં પંચાયત અને નિગમની ચૂંટણી પહેલાં આતંકીઓ તરફથી નેતાઓને ટાર્ગેટ કરવામાં આવી રહ્યા છે
transliteration	Jam'mu-kāśmīramāṁ pan̄cāyata anē nigamanī cūṇṭaṇī pahēlāṁ ātaṅkī'ō taraphathī nētā'ōnē ṭārgēṭa karavāmāṁ āvī rahyā chē
reference	Leaders are being targeted by the terrorists prior to the Panchayat and Municipal elections in Jammu and Kashmir.
Transformer	The leadership of the development and 33 election before the 1965 and 33 election before the party
Multilingual Transformer+Averaging	In the first elections to the Jammu - Kashmir panchayat and the Corporation, leaders are being pushed through terrorism.
source	જોકે, ભારતીય જનતા પાર્ટી અહીં ચૂંટણી લડી રહી છે.
transliteration	Jōkē, bhāratīya janatā pārṭī ahīṁ cūṇṭaṇī laḍī rahī chē.
reference	However, the Bhartiya Janata Party is contesting the elections there.
Transformer	However, the indian janata party is elected in 2017.
Multilingual Transformer+Averaging	However, the Bharatiya Janata Party has been elections held here.
source	તેઓ પોતાના ઘરે પરત જઈ રહ્યાં હતાં.
transliteration	Tē'ō pōtānā gharē parata ja'ī rahyāṁ hatāṁ.
reference	He was returning to his residence.
Transformer	He returned to his home.
Multilingual Transformer+Averaging	He was going to his home.
source	જોકે, નિરાશ થવાની જરૂર નથી હવે ફરી એકવાર ભારત અને પાકિસ્તાની ટીમો આમને સામને ટકરાશે.
transliteration	Jōkē, nirāśa thavānī jarūra nathī havē pharī ēkavāra bhārata anē pākistānī ṭīmō āmanē sāmanē ṭakarāśē.
reference	However, there is no need to be frustrated. India and Pakistan teams will face each other once again.
Transformer	However, not authentication, once it will be not authentication and the Pakistani teams will not be killed.
Multilingual Transformer+Averaging	However, there is no need to disappoint, and again the teams of Indian and Pakistani teams clashed with each other.

5.6 Summary

We believe that NMT is a promising approach for Machine Translation for low resource languages. But we need various techniques to handle the data scarcity problem. Transfer Learning and Multilingual Machine Translation are two important areas of research that tackles this problem. In this chapter, we showed that how Multilingual MT models are more effective than the individually trained MT models for a low resource language pair. We presented our results on the Gujarati→English language pair and achieved significant BLEU improvements.

Chapter 6

LTRC-MT Simple & Effective Hindi-English Neural Machine Translation Systems at WAT 2019

Hindi is the fourth most commonly spoken language in the world with an estimated reach of 500 million people in the Indian subcontinent. At the same time, it is a low resourced language. This chapter describes our work on Neural Machine Translation for Hindi to English using Recurrent Neural Networks and Transformer architecture. The baseline NMT do not yield acceptable translation quality due to limited training data. However, the use of synthetic parallel data (generated using back translation, based on an NMT baseline) significantly improves translation quality. Our best performing translation system ranked as the runner-up amongst all the systems that participated in the Hindi to English translation task as a part of Workshop on Asian Translation (WAT) @ EMNLP-IJCNLP 2019.

6.1 Introduction

Neural Machine Translation [30, 19, 31, 32, 1] has been receiving considerable attention in the recent years, given its superior performance without the demand of heavily hand crafted engineering efforts. NMT often outperforms Statistical Machine Translation (SMT) techniques but it still struggles if the parallel data is insufficient like in the case of Indian languages. Hindi being one of the most common spoken Indian languages, continue to remain as a low resource language demanding further attention from the research community. The Hindi-English pair has limited availability of sentence level aligned bitext as parallel corpora.

This chapter describes an overview of the submission of our team (IIIT Hyderabad (LTRC)) in WAT 2019 [59] Hindi-English Machine Translation shared task. We experimented with both attention-based LSTM encoder-decoder architecture & the recently proposed Transformer architecture. We used Byte Pair Encoding (BPE) to enable open vocabulary translation. We then leveraged synthetic data generated by our own models to improve the translation performance.

6.1.1 Subword Segmentation for NMT

Neural Machine Translation relies on first mapping each word into the vector space, and traditionally we have a word vector corresponding to each word in a fixed vocabulary. Addressing the problem of data scarcity and the hardness of the system to learn high quality representations for rare words, [38] proposed to learn subword units and perform translation at a subword level. With the goal of open vocabulary NMT, we incorporate this approach in our system as a preprocessing step. In our early experiments, we note that Byte Pair Encoding (BPE) works better than UNK replacement techniques & also aids in better translation performance. For all of our systems, we learn separate vocabularies for Hindi and English each with 32k merge operations. With the help of BPE, the vocabulary size is reduced drastically and we no longer need to prune the vocabularies. After the translation, we do an extra post processing step to convert the target language subword units back to normal words. We found this approach to be very helpful in handling rare word representations.

6.1.2 Synthetic Training Data

To utilize monolingual data along with IITB corpus, we employ back translation. Backtranslation [53] is a widely used data augmentation technique for aiding Neural Machine Translation for languages low on parallel data. The method works by generating synthetic data on the source side from target side monolingual data using a target-to-source NMT model. The synthetic parallel data thus formed is combined with the actual parallel data to train a new NMT model. We used around 10M English sentences and backtranslated them into Hindi using a English-Hindi NMT model.

6.2 Experimental Setup

6.2.1 Dataset

In our experiments, we used IIT-Bombay [39] Hindi-English parallel data provided by the organizers. The training corpus provided by the organizers, consists of data from mixed domains. There are roughly 1.5M samples in training data from diverse sources, while the development and test sets are from news domains. In addition to this, around 10M English monolingual data from WMT14 newscrawl articles is used in our backtranslation enabled attempts at training an NMT system.

Table 6.1: Statistics of our processed parallel data.

Dataset	Sentences	Tokens
IITB Train	15,28,631	21.5M / 20.3M
IITB Test	2,507	62.3k / 55.8k
IITB Dev	520	9.7k / 10.3k

6.2.2 Data Processing

We used Moses [40] toolkit for tokenization and cleaning the English side of the data. Hindi side of the data is first normalized with Indic NLP library[1] followed by tokenization with the same library. As our preprocessing step, we removed all the sentences of length greater than 80 from our training corpus. We used BPE segmentation with 32k merge operations. During training, we lowercased all of our training data & used truecase[2] as a truecaser during testing.

6.2.3 Training Details

For all of our experiments, we used OpenNMT-py [41] toolkit. We used both attention-based LSTM models and Transformer models in our submissions.

We used an LSTM based Bi-directional encoder and a unidirectional decoder along with global attention mechanism. We kept 4 layers in both the encoder & decoder with embedding size set to 512. The batch size was set to 64 and a dropout rate of 0.3. We used Adam optimizer [58] for all our experiments.

For our transformer model, we used 6 layers in both encoder and decoder with 512 hidden units in each layer. The word embedding size was set to 512 with 8 heads. The training is run in batches of maximum 4096 tokens at a time with dropout set to 0.3. The model parameters are optimized using Adam optimizer.

6.3 Results and Discussion

In table 6.2, we report case-sensitive Bilingual Evaluation Understudy (BLEU) [22] score, Rank-based Intuitive Bilingual Evaluation Score (RIBES) [60], Adequacy-fluency metrics (AM-FM) [61] and the Human Evaluation results provided by WAT 2019 for all our attempts. The results show that our NMT system based on Transformer & backtranslation is ranked 2nd among all the constraint submissions made in WAT 2019 Hindi-English shared task & is ranked 3rd overall. In the human evaluation, our best performing system got a rating of 3.43 out of 5.

[1] https://anoopkunchukuttan.github.io/indic_nlp_library/
[2] https://pypi.org/project/truecase/

Table 6.2: This table describes the results of WAT 2019 evaluation of our submitted systems & compared with the best system submissions of WAT 2019 & the previous year. 'BT' stands for backtranslation.

Architecture	BLEU	RIBES	AM-FM	Human
Transformer	16.32	0.729072	0.563590	-
LSTM with global attention + BT	17.07	0.729059	0.587060	-
Transformer + BT	**18.64**	**0.735358**	**0.594770**	**3.43**
2018 Best	17.80	0.731727	0.611090	2.96
2019 Best (Constraint)	19.06	0.741197	0.566490	3.83
2019 Best (Unconstraint)	22.91	0.768324	0.641730	4.14

We also manually inspected the translation samples generated by our baseline and backtranslation models. We found that the translations generated by the backtranslation models captures more information in general and are more fluent and meaningful than the translations generated by the baseline model. In table 6.3, we have provided some translation outputs generated by our baseline model and compared it with the outputs generated by our backtranslation models.

Table 6.3: Hindi-English Translation examples illustrating the quality of our baseline and best performing models (backtranslation) submitted at WAT@EMLP-IJCNLP 2019 shared task.

system	sentence
source	मेहमानों को लाने - ले जाने के लिए 32 चार्टर्ड विमानों की व्यवस्था की गई है।
transliteration	mehamaanon ko laane - le jaane ke lie 32 chaartard vimaanon kee vyavastha kee gaee hai.
reference	32 chartered planes have been booked to ferry the guests to and fro.
Transformer	32 chartered aircraft have been provided for bringing the guests.
Transformer+BT	Thirty-two chartered aircrafts have been arranged to transport guests.
source	एल . एन . मित्तल , सचिन तेंदुलकर , फिल्म अभिनेता अनिल कपूर , आमिर खान , ए . आर . रहमान गुरुवार शाम ही जोधपुर पहुंचे गए।
transliteration	el . en . mittal , sachin tendulakar , philm abhineta anil kapoor , aamir khaan , e . aar . rahamaan guruvaar shaam hee jodhapur pahunche gae.
reference	L.N. Mittal, Sachin Tendulkar, and bollywood actors Anil Kapoor, Aamir Khan and A.R. Rahman has already arrived in Jodhpur on thursday evening.
Transformer	L.N. Mittal, Sachin Tendulkar, film actor Anil Kapoor, Aamir Khan, A.R. Rahman went to Jodhpur in the evening.
Transformer+BT	L. N. Mittal, Sachin Tendulkar, film actor Anil Kapoor, Aamir Khan, A.R. Rahman arrived in Jodhpur just thursday evening.
source	उन्होंने कहा कि यूरोपीय नेताओं को अपने लोगों के साथ , वर्षों तक इस्तेमाल किए गए अपने जासूसी कार्यक्रमों के प्रति भी ईमानदार होना पड़ेगा।
transliteration	unhonne kaha ki yooropeey netaon ko apane logon ke saath , varshon tak istemaal kie gae apane jaasoosee kaaryakramon ke prati bhee eemaanadaar hona padega.
reference	He said European leaders need to be honest with their own people about the kind of espionage programs they've used for years themselves.
Transformer	He said the european leaders had to be faithful to their provincial programmes for years with their people.
Transformer+BT	He said European leaders would also have to be honest with their people, with their own zumqi programmes used for years.
source	अयोध्या नगर निवासी नरेंद्र बहाड़ (21) की इलेक्ट्रानिक्स की दूकान है।
transliteration	ayodhya nagar nivaasee narendr bahaad (21) kee ilektraaniks kee dookaan hai.
reference	Narendra Bahad (21), a resident of Ayodhya city, is the owner of an electronics shop.
Transformer	The inhabitants of Ayodhya city , Narendra Bahad the power of electronics.
Transformer+BT	Ayodhya city resident Narendra Bahad (21) has a dukan of electronics.

6.4 Summary

We believe that NMT is indeed a promising approach for Machine Translation of low resource languages. In this chapter, we showed the effectiveness of Transformer models on a low resource language pair Hindi-English. Additionally, we show how synthetic data can help improving the NMT systems for Hindi-English.

Chapter 7

A2C-NMT: A Reinforcement Learning Approach for Neural Machine Translation

Recent studies have shown that Reinforcement Learning is an effective approach to improve the performance of NMT systems. In this chapter, we present an approach for training Neural Machine Translation systems using Advantage Actor-Critic method from Reinforcement Learning. Our approach directly optimizes the model parameters with respect to the task-specific scores, unlike conventional maximum likelihood estimation and is fit for problems with low resource settings, large action space & delayed rewards. We also demonstrate experiments to leverage our approach to further boost the performance of NMT systems using source & target monolingual data for a low resource language pair. On German-English & English-French translation tasks, our approach yields +1.26 to 1.38 BLEU improvements over the strong RL baselines for NMT.

7.1 Introduction

Neural Machine Translation [30, 19, 31, 32, 1] has been receiving considerable attention in the recent years, given its superior performance without the demand of heavily hand crafted engineering efforts. NMT models are usually trained using Maximum Likelihood Estimation (MLE) [62] objective. Although MLE is easy to implement, the token-level objective function during training is inconsistent with the sequence level evaluation metrics such as BLEU [22], as argued in [23]. Such an objective does not guarantee the translation results to be natural, sufficient, and accurate compared with ground-truth translation by humans.

To address this problem, much recent work attempts to reduce the inconsistency between training and inference, such as adopting sequence level objectives and directly optimizing BLEU scores [63, 64, 65, 66, 67, 68].

Yet somewhat improved, the previous RL methods for NMT have some limitations. Most of the works [63, 64, 68] employ REINFORCE [69] algorithm which is a Naive Monte Carlo

Reinforcement Learning algorithm that estimates the Q values by sampling and yields very high variance when the action space is large, which is a serious problem for NLP tasks like Machine Translation. [65] proposed an actor-critic algorithm for sequence prediction which uses a Q critic model for gradient approximation and showed that actor-critic algorithm outperformed REINFORCE algorithm. The critic model in their setting needs reference translations during reinforcement training and the reward function is known and can be exploited to compute immediate rewards after taking each action. However, practically, the rating or reward for a translation is only provided once it is completed. [65] reports that their algorithm degrades if the rewards are delayed.

We, in this work, adopt a different method for training and improving NMT models, targeting at directly minimizing the difference between human/reference translations and the translations given by the NMT model. To achieve this target, inspired by the recent success of Advantage Actor-Critic algorithm [70], we propose a new Reinforcement Learning approach for NMT and name it as A2C-NMT. In A2C-NMT, we combine the Advantage Actor-Critic algorithm with the attention-based neural encoder-decoder architecture. A2C-NMT is better suited for problems with large action space like NMT and has many advantages over the actor-critic implementation in [65]. Unlike in [65], the critic model in our setting does not need ground truth translations to compute the gradient during Reinforcement training. Rather it just needs the reward for a translation of a sample drawn from the actor model which also makes it fit for unsupervised Machine Translation. A2C-NMT is also naturally fit for problems in real world scenarios like translation rating interfaces, where the users can be asked to provide ratings for the machine-generated translations and then the user ratings can be used to improve the model. Our experiments on standard translation tasks demonstrates that A2C-NMT achieves better translation results than traditional attention-based encoder-decoder NMT model and also outperforms some competitive RL baselines.

We also leverage our approach with source & target monolingual data for enhancing the NMT model performance for languages where parallel data is insufficient and demonstrate our experiments on a low resource language pair Hindi-English and achieve high BLEU score.

7.2 Reinforcement Learning architecture of A2C-NMT

In this section, we introduce the key ideas of our approach and present an algorithm for NMT using Reinforcement Learning.

7.2.1 Formulation of NMT model as an MDP

A basic reinforcement learning problem can be modeled as a Markov Decision Process (MDP) by defining the environment and states, agent's actions, reward function and transition prob-

abilities. We model the Neural Machine Translation task as a Markov Decision Process which operates on continuous state space. The action space for our MDP is the target language's vocabulary and the states are the hidden vectors h_t^{dec} generated by the decoder.

The NMT model starts with an initial stage h_0^{dec} which is a representation of the source sentence x computed by the encoder. At any time step $t > 0$, the model decides the next action to take by defining a stochastic policy $P_\theta(y_t|y_{<t}, x)$, which takes the previous hidden state vector h_{t-1}^{dec} as input and produces a probability distribution over all actions which are defined by the tokens in the target language vocabulary. The next action $\hat{y}_t$ is chosen either by sampling from this policy or by taking $argmax$. The model computes the next state h_t^{dec} by updating the current state h_{t-1}^{dec} by the action taken $\hat{y}_t$.

7.2.2 Policy Gradient Method for our NMT model

The objective of our RL framework is to find a policy that maximizes the quality of translations sampled from our model's (actor) policy:

$$\max_\theta J_{pg}(\theta) = \max_\theta \mathrm{E}_{\substack{x\sim y \\ \hat{y}\sim P_\theta(.|x)}} [R(\hat{y}, x)] \tag{7.1}$$

where R is the reward function that returns a score in $[0, 1]$ reflecting the quality of translation of the input x. In this work we use sentence-level BLEU score as our reward function & optimize this objective function $J(\theta)$ with policy gradient method. But one can easily change the reward function & use such metrics which are only dependent on source sentence & the hypothesis thereby making an unsupervised model. For a fixed x, the gradient of the objective in Eq. 7.1 is:

$$\begin{aligned} \nabla_\theta J_{pg}(\theta) &= \mathrm{E}_{\hat{y}\sim P_\theta(.)}[R(\hat{y})\nabla_\theta log P_\theta(\hat{y})] \\ &= \textstyle\sum_{t=1}^{m} \mathrm{E}_{\hat{y}_t\sim P(.|\hat{y}_{<t})}[R(\hat{y})\nabla_\theta \log P_\theta(\hat{y}_t|\hat{y}_{<t})] \end{aligned} \tag{7.2}$$

7.2.3 Advantage Actor-Critic

Algorithm 1 The A2C algorithm for NMT.

1: pretrain actor with MLE objective
2: pretrain critic for 5 epochs
3: **for** $k = 0...K$ **do**
4: receive a source sentence x
4: sample a translation: $\hat{y} \sim P_\theta(y|x)$
5: receive a reward $R(\hat{y}, x)$
6: update the actor model using the gradient in Eq. 6.3
7: update the critic model using the gradient in Eq. 6.5
8: **end for**=0

In order to compute the gradient in Eq. 7.2, the system should provide reward values for each possible translation. This is not feasible using Naive Monte Carlo Reinforcement Learning

algorithms such as REINFORCE [69] which estimates the Q values (expected future rewards of a sample translation $\hat{y}$) by sampling and yields very high variance when the action space is large, leading to training instability.

We thus employ the approach of advantage actor-critic (A2C) algorithm [70], which combines the REINFORCE [69] algorithm with actor-critic. The algorithm approximates the gradient in Eq. 7.2 by a single-point sample and normalize the rewards by V values to reduce variance:

$$\begin{gathered}\nabla_\theta J_{pg}(\theta) \approx \sum_{t=1}^{m} \nabla_\theta \log P_\theta(\hat{y}_t|\hat{y}_{<t}, x) R'_t(\hat{y}_{<t}, x) \\ \text{with } R'_t(\hat{y}_{<t}, x) \equiv R(\hat{y}, x) - V(\hat{y}_{<t}, x)\end{gathered} \tag{7.3}$$

where $V(\hat{y}_{<t}, x) = \mathrm{E}[R(\hat{y}, x)|\hat{y}_{<t}, x]$ is a baseline that estimates the expected future reward given x and $\hat{y}_{<t}$.

We train a separate critic model to estimate the V values. This model is also an attention-based encoder-decoder model [30] that encodes a source sentence x and decodes a predicted translation $\hat{y}$. At time step t, it computes $V_\omega = W_o \tilde{h}_t^{dec}$ where $\tilde{h}_t^{dec}$ is the hidden state of the RNN decoder, and W_o is a matrix that transforms a vector into a scalar. The V values computed by the critic model are used to approximate the gradient of the expected reward with respect to the parameters of the actor model (main sequence prediction network). The critic model is trained to minimize the Mean Square Error between its estimates and the truth values:

$$J_{crt}(\omega) = \mathrm{E}_{x\sim D}\left[\sum_{t=1}^{m} \|R(\hat{y}, x) - V_\omega(\hat{y}_{<t}, x)\|^2\right] \tag{7.4}$$

Algorithm 1 summarizes our A2C algorithm. For each x , we draw a single sample translation $\hat{y}$ from the actor model and then receive a reward $R(\hat{y}, x)$ from the environment, which is used for both estimating the gradient of the NMT model (actor) (Eq. 7.3) and the gradient of the critic model with respect to ω:

$$\nabla_\omega J_{crt}(\omega) = \sum_{t=1}^{m} [R(\hat{y}_{<t}, x) - V_\omega(\hat{y}_{<t})] \nabla_\omega V_\omega(\hat{y}_{<t}) \tag{7.5}$$

7.3 Experiments

We report our RL training results on two translation tasks German→English (De→En) and English→French (En→Fr). We report our results of RL training with monolingual data on a low resource language pair Hindi→English (Hi→En).

Dataset: For En→Fr translation, we use the same subset of WMT14 corpus as training set to have a fair comparison with previous works [64, 44]. We use newstest2012 & newstest2013 as development set and newstest2014 as test set. The training data consists of approximately 12M sentence pairs. Maximal sentence length is set to 50. We use Moses [40] toolkit for tokenization and cleaning the data.

For De→En translation, to compare with the previous works [63, 65], we use the dataset from IWSLT 2014 evaluation campaign [71], consisting of training/dev/test corpus with approximately 153k, 7k and 6.5k bilingual sentence pairs respectively. The maximal sentence length is also set as 50.

For Hi→En translation, we make use of the largest parallel corpora available for Hindi-English: IIT-Bombay parallel corpus [39]. It consists of approximately 1.4 million parallel sentence pairs. We use the development set provided with the corpus and test our system on WAT2017 Hindi-English test-set. For our experiments on monolingual data with RL training, we use Hindi monolingual corpus provided with the IIT-Bombay parallel corpus and NewsCrawl articles from 2007 & 2010 (WMT14) as our English monolingual corpus. The Hindi side of the corpus is first normalized with Indic NLP library[1] followed by tokenization with the same library.

Subword Segmentation for NMT: Neural Machine Translation relies on first mapping each word into the vector space, and traditionally we have a word vector corresponding to each word in a fixed vocabulary. Addressing the problem of data scarcity and the hardness of the system to learn high quality representations for rare words, [38] proposed to learn subword units and perform translation at a subword level. With the goal of open vocabulary NMT, we incorporate this approach in our system as a preprocessing step. In our early experiments, we note that Byte Pair Encoding (BPE) works better than UNK replacement techniques. For our experiments on German→English and French→English, we learn a shared word-peice vocabulary with 32k merge operations. For Hindi-English experiments, we learn seperate vocabularies for Hindi and English each with 32k BPE merge operations. With the help of BPE, the vocabulary size is reduced drastically and we no longer need to prune the vocabularies. After the translation, we do an extra post processing step to convert the target language subword units back to normal words. We find this approach to be very helpful in handling rare word representations.

Implementation Details: In A2C-NMT, the structure of the NMT model (actor) is same as in [30], an LSTM based encoder-decoder framework with Global Attention Mechanism. We use a Bi-directional encoder and a unidirectional decoder with 4 layers. We used Adam optimizer [58] for all the tasks. Embedding dimensionality is set to 512 and batch size is 64.

The critic model is our setting follows the same architecture of the actor model. For the training of our A2C-NMT model, we first pre-train the actor model using MLE objective with a batch size of 64 and optimize it using Adam optimizer. Then, we pretrain the critic model for about 5 epochs using the data $(x, \hat{y})$, sampled from the actor model and using the ground truth translations (x, y). After that, the training enters the Reinforcement Learning mode, in which ground truth translations are not available to the critic for gradient estimation. For each source sentence x, we draw a single sample translation $\hat{y}$ from the actor model and is used in estimating

[1] https://anoopkunchukuttan.github.io/indic_nlp_library/

the gradients of both actor and the critic model. We run our algorithm in mini-batches of x and aggregates the gradient over all x in a mini-batch for each update. We use a learning rate of 0.001 during pre-training and 0.0001 during reinforcement learning mode. During pre-training mode, we decay the learning rate by a factor of 0.5 when the perplexity on the development set increases. During RL training our model learns in an unsupervised setting, where ground truth translations are not available. We used Per-sentence BLEU: average sentence-level BLEU score of translations sampled and scored during reinforcement learning mode as our reward function for the policy gradient method in our NMT model.

7.4 Results and Discussion

In table 7.1, we provide the En→Fr translation results together with several strong baselines. To make our comparison comprehensive, we compare our results with the other published works which use other training objectives rather than MLE [64, 44]. Since, we use the method of subword neural machine translation [38], we also compare our results with [32], in which the authors presented an adversarial training approach for NMT.

From the table, we can clearly observe that A2C-NMT obtains satisfactory translation quality against baseline systems. For pre-training the NMT model, we use bidirectional LSTM with global attention mechanism and greedy search for decoding, resulting in a BLEU score of 32.24. After Reinforcement training, the model improves by significant margin of +1.62 and results in BLEU score of 33.86.

Table 7.1: Comparison with previous work on En→Fr translation task.

System	**BLEU**
MRT [64]	31.30
Adversarial-NMT [32]	31.91
Dual Learning [44]	32.06
NMT baseline	32.24
A2C-NMT	**33.86**

In table 7.2, we provide the De→En translation results compared with several strong RL baselines. We also compared our systems' performance with [72] which employs beam search for decoding. Our NMT baseline system obtains a BLEU score of 26.67. After Reinforcement Training, our model improves by significant margin of +1.69, resulting in a BLEU score of 28.36.

In table 7.3, we provide the Hi→En translation results together with some baselines presented at WAT2017. Our baseline system achieves a BLEU score of 16.32 and after Reinforcement training it results in a BLEU score of 18.17 showing an increase of +1.85 BLEU points. We randomly pick 2.5M sentences each from the pool of Hindi & English monolingual data and

Table 7.2: Comparison with previous work on IWSLT2014 German-English translation task.

System	BLEU
MIXER [63]	21.83
AC [65]	22.45
MRT [64]	25.84
BSO [72]	26.36
Adversarial-NMT [32]	26.98
NMT baseline	26.67
A2C-NMT	**28.36**

Table 7.3: Comparison with previous work on WAT2017 Hi→En translation task.

System	BLEU
CNNS2S [73]	13.76
Transformer (CUNI)	17.80
Ensemble [74]	22.44
NMT baseline	16.32
A2C-NMT	18.17 (+1.85)
NMT+Monolingual data	21.28 (+4.96)
A2C-NMT+Monolingual data	**22.80 (+6.48)**

translate them to get a pseudo parallel corpus. Training a NMT model on the combined data with MLE objective, results in a BLEU score of 21.28. After RL training, the model resulted in a BLEU score of 22.80.

7.5 Related Work

Reinforcement learning has become a standard technique for incorporating feedback across diverse tasks such as virtual assistants [75] and robot voice control [76]. There has been an increasing interest in applying reinforcement learning to a variety of problems but only a few studies address problems with natural language actions spaces. In natural language processing tasks, reinforcement learning systems have been applied to dialogue management systems [77], extracting textual knowledge to improve game control performance [78] and learning to translate in real time with NMT [79].

Our work is mainly related with the literature of using reinforcement learning to directly optimize sequence level metrics for neural machine translation. Some of the representative works are [63, 64, 65]. In [63], the authors propose to train a neural encoder-decoder model for translation with the objective gradually shifting from maximizing token-level likelihood to optimizing sentence-level BLEU score. This approach is closely related to ours but requires a

policy-mixing strategy and only uses a linear critic model. [64] adopts minimum risk training to minimize task-specific expected loss that is induced by BLEU score on NMT training data. Both of the above works employs REINFORCE [69] algorithm which is a Naive Monte Carlo Reinforcement Learning algorithm that estimates the Q values by sampling and yields very high variance when the action space is very large. Further, [65] optimizes the policy by employing actor-critic algorithm. [20] also introduces a simple RL based method to optimize the stacked LSTM model for NMT. The algorithm used in [80] is similar to ours but they use the algorithm in a Bandit Learning task where the model is improved by user ratings. Whereas in our approach we leverage the algorithm to build effective NMT system for Hindi-English via utilizing monolingual data and also provide comparisons with other RL techniques on standard datasets.

Our work is also related to works that leverage monolingual data for improving NMT models [53, 81, 44, 82]. [81] exploits source monolingual data in NMT. [53] proposes a back-translation method to leverage target-side monolingual data for NMT. [82] proposes a semi-supervised approach for NMT. [44] formulates machine translation as a communication game, which leverages the power of two-directional translational models and source/target monolingual data. Although, the previous works have used monolingual data for NMT training but none of these works explored the power of monolingual data in the context of RL training.

7.6 Summary

We have presented a RL approach to effectively train NMT models which can directly optimize sentence level metrics such as BLEU and is also fit for real world scenarios to incorporate feedback on machine generated translations. On German-English and English-French translation tasks, it outperforms several RL baselines significantly. We also leverage our RL approach with monolingual data to build a strong Hindi-English NMT system. In future, we aim to leverage our algorithm with self-attention based Transformer [1] model along with abundant monolingual data for Unsupervised NMT.

Chapter 8

Conclusions Future Work

In this work, we investigated whether the linguistic input features are helpful in improving the translation performance of the state-of-the-art Transformer based NMT model, and our empirical results show that this is the case. We presented our results on Hindi-English, a low resource language pair where Hindi is a morphologically rich and a free word order language whereas on the other end we have English which is morphologically less complicated and word order specific language. We empirically tested the inclusion of various linguistic features, including lemmas, part-of-speech tags, morphological features and IOB tags for a (sub)word model. Our experiments showed that linguistic input features yield significant improvements over both (sub)word based NMT baseline.

We explored effective methods to exploit parallel data from multiple related languages to improve the translation between Indian languages and English. We presented a technique to leverage language similarity between related languages and also proposed a new Multilingual Transfer Learning approach that outperforms the original Transfer Learning approach by significant BLEU points. We presented our Hindi-English Neural MT system and showed how synthetic data can help in improving the NMT systems for Hindi-English. We also explored the power of training multilingual models using relatively high resourced related language pair like Hindi-English to assist the translation of a low resource language pair such as Gujarati-English.

We have also presented a Reinforcement Learning approach to effectively train NMT models which can directly optimize the sentence level evaluation metrics such as BLEU. On German-English and English-French translation tasks, it outperformed several RL baselines significantly. We also demonstrated experiments to leverage our approach to further boost the performance of NMT systems using source and target monolingual data for a low resource language pair like Hindi-English.

In this thesis, we addressed the three problems in Indian Language translation 1) difficulty in translation of Indian languages due to it's complex nature (chapter 3); 2) data scarcity problem (chapter 4) and 3) training and testing inconsistency in NMT (chapter 7). Overall, we have demonstrated various techniques that can be used to build efficient neural machine

translation systems for Indian Languages and we hope we have made significant contribution in the research on Indian language NMT.

Related Publications

1. **Vikrant Goyal**, Sourav Kumar, Dipti Misra Sharma, "Efficient Neural Machine Translation for Low Resource Languages via Leveraging Related Languages," accepted in the *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020): Student Research Workshop.*

2. **Vikrant Goyal**, Pruthwik Mishra, Dipti Misra Sharma, "Linguistically Informed Hindi-English Neural Machine Translation," accepted in the *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020).*

3. **Vikrant Goyal** and Dipti Misra Sharma, "LTRC-MT Simple & Effective Hindi-English Neural Machine Translation Systems at WAT 2019," accepted in the *Proceedings of the 6th Workshop on Asian Translation (WAT)*, 2019.

4. **Vikrant Goyal** and Dipti Misra Sharma, "The IIIT-H Gujarati-English Machine Translation system for WMT19," accepted in the *Proceedings of the Fourth Conference on Machine Translation (WMT)* ,2019.

Bibliography

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[2] B. Zoph, D. Yuret, J. May, and K. Knight, "Transfer learning for low-resource neural machine translation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1568–1575.

[3] R. H. Richens, "Interlingual machine translation," *The Computer Journal*, vol. 1, no. 3, pp. 144–147, 1958.

[4] I. K. Bel'skaja, "Machine translation of languages," *Research (London)*, vol. 10, pp. 383–389, 1957.

[5] R. Sinha and A. Jain, "Anglahindi: an english to hindi machine-aided translation system," *MT Summit IX, New Orleans, USA*, pp. 494–497, 2003.

[6] F. Wong, M. Dong, and D. Hu, "Machine translation using constraint-based synchronous grammar," *Tsinghua Science and Technology*, vol. 11, no. 3, pp. 295–306, 2006.

[7] S. Nirenburg, J. Carbonell, M. Tomita, and K. Goodman, *Machine translation: A knowledge-based approach*. Morgan Kaufmann Publishers Inc., 1994.

[8] A. Kruger, K. Wallmach, and J. Munday, *Corpus-based translation studies: Research and applications*. Bloomsbury Publishing, 2011.

[9] O. Dhariya, S. Malviya, and U. S. Tiwary, "A hybrid approach for hindi-english machine translation," in *2017 International Conference on Information Networking (ICOIN)*. IEEE, 2017, pp. 389–394.

[10] Y. Al-Onaizan, J. Curin, M. Jahr, K. Knight, J. Lafferty, D. Melamed, F.-J. Och, D. Purdy, N. A. Smith, and D. Yarowsky, "Statistical machine translation," in *Final Report, JHU Summer Workshop*, vol. 30, 1999.

[11] A. Lopez, "Statistical machine translation," *ACM Computing Surveys (CSUR)*, vol. 40, no. 3, pp. 1–49, 2008.

[12] P. Koehn, *Statistical machine translation*. Cambridge University Press, 2009.

[13] M. Galley and C. D. Manning, "A simple and effective hierarchical phrase reordering model," in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008, pp. 848–856.

[14] D. Chiang, K. Knight, and W. Wang, "11,001 new features for statistical machine translation," in *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics*, 2009, pp. 218–226.

[15] S. Green, S. I. Wang, D. Cer, and C. D. Manning, "Fast and adaptive online training of feature-rich translation models," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2013, pp. 311–321.

[16] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1700–1709.

[17] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[18] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[20] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," *arXiv preprint arXiv:1609.08144*, 2016.

[21] P. Koehn and R. Knowles, "Six challenges for neural machine translation," *arXiv preprint arXiv:1706.03872*, 2017.

[22] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[23] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 1171–1179.

[24] R. Sinha, K. Sivaraman, A. Agrawal, R. Jain, R. Srivastava, and A. Jain, "Anglabharti: a multilingual machine aided translation project on translation from english to indian languages," in *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, vol. 2. IEEE, 1995, pp. 1609–1614.

[25] S. Chand, "Empirical survey of machine translation tools," in *2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*. IEEE, 2016, pp. 181–185.

[26] P. Antony, "Machine translation approaches and survey for indian languages," in *International Journal of Computational Linguistics & Chinese Language Processing, Volume 18, Number 1, March 2013*, 2013.

[27] A. Kunchukuttan, A. Mishra, R. Chatterjee, R. Shah, and P. Bhattacharyya, "Sata-anuvadak: Tackling multiway translation of indian languages," *pan*, vol. 841, no. 54,570, pp. 4–135, 2014.

[28] R. Agrawal, M. Shekhar, and D. M. Sharma, "Three-phase training to address data sparsity in neural machine translation," in *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, 2017, pp. 13–22.

[29] R. Agrawal and D. M. Sharma, "Experiments on different recurrent neural networks for english-hindi machine translation."

[30] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[31] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado *et al.*, "Google's multilingual neural machine translation system: Enabling zero-shot translation," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 339–351, 2017.

[32] L. Wu, Y. Xia, L. Zhao, F. Tian, T. Qin, J. Lai, and T.-Y. Liu, "Adversarial neural machine translation," *arXiv preprint arXiv:1704.06933*, 2017.

[33] R. Sennrich and B. Haddow, "Linguistic input features improve neural machine translation," *arXiv preprint arXiv:1606.02892*, 2016.

[34] J. Niehues and E. Cho, "Exploiting linguistic resources for neural machine translation using multi-task learning," *arXiv preprint arXiv:1708.00993*, 2017.

[35] Q. Li, D. F. Wong, L. S. Chao, M. Zhu, T. Xiao, J. Zhu, and M. Zhang, "Linguistic knowledge-aware neural machine translation," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 26, no. 12, pp. 2341–2354, 2018.

[36] A. Alexandrescu and K. Kirchhoff, "Factored neural language models," in *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*. Association for Computational Linguistics, 2006, pp. 1–4.

[37] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013, pp. 746–751.

[38] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.

[39] A. Kunchukuttan, P. Mehta, and P. Bhattacharyya, "The iit bombay english-hindi parallel corpus," *arXiv preprint arXiv:1710.02855*, 2017.

[40] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens *et al.*, "Moses: Open source toolkit for statistical machine translation," in *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, 2007, pp. 177–180.

[41] G. Klein, Y. Kim, Y. Deng, V. Nguyen, J. Senellart, and A. M. Rush, "Opennmt: Neural machine translation toolkit," *arXiv preprint arXiv:1805.11462*, 2018.

[42] P. Koehn and H. Hoang, "Factored translation models," in *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, 2007, pp. 868–876.

[43] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 86–96.

[44] D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T.-Y. Liu, and W.-Y. Ma, "Dual learning for machine translation," in *Advances in Neural Information Processing Systems*, 2016, pp. 820–828.

[45] G. Lample, A. Conneau, L. Denoyer, and M. Ranzato, "Unsupervised machine translation using monolingual corpora only," *arXiv preprint arXiv:1711.00043*, 2017.

[46] M. Artetxe, G. Labaka, E. Agirre, and K. Cho, "Unsupervised neural machine translation," *arXiv preprint arXiv:1710.11041*, 2017.

[47] G. Lample, M. Ott, A. Conneau, L. Denoyer, and M. Ranzato, "Phrase-based & neural unsupervised machine translation," *arXiv preprint arXiv:1804.07755*, 2018.

[48] T. Kocmi and O. Bojar, "Trivial transfer learning for low-resource neural machine translation," *arXiv preprint arXiv:1809.00357*, 2018.

[49] O. Firat, K. Cho, B. Sankaran, F. T. Yarman Vural, and Y. Bengio, "Multi-way, multilingual neural machine translation," *Computer Speech and Language*, vol. 45, no. C, pp. 236–252, 2017.

[50] P. Passban, Q. Liu, and A. Way, "Translating low-resource languages by vocabulary adaptation from close counterparts," *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, vol. 16, no. 4, pp. 1–14, 2017.

[51] T. Q. Nguyen and D. Chiang, "Transfer learning across low-resource, related languages for neural machine translation," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2017, pp. 296–301.

[52] G. N. Jha, "The tdil program and the indian langauge corpora intitiative (ilci)." in *LREC*, 2010.

[53] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," *arXiv preprint arXiv:1511.06709*, 2015.

[54] T.-L. Ha, J. Niehues, and A. Waibel, "Toward multilingual neural machine translation with universal encoder and decoder," *arXiv preprint arXiv:1611.04798*, 2016.

[55] O. Firat, K. Cho, and Y. Bengio, "Multi-way, multilingual neural machine translation with a shared attention mechanism," *arXiv preprint arXiv:1601.01073*, 2016.

[56] D. Dong, H. Wu, W. He, D. Yu, and H. Wang, "Multi-task learning for multiple language translation," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, 2015, pp. 1723–1732.

[57] B. Zoph and K. Knight, "Multi-source neural translation," *arXiv preprint arXiv:1601.00710*, 2016.

[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[59] T. Nakazawa, C. Ding, R. Dabre, H. Mino, I. Goto, W. P. Pa, N. Doi, Y. Oda, A. Kunchukuttan, S. Parida, O. Bojar, and S. Kurohashi, "Overview of the 6th workshop on Asian translation," in *Proceedings of the 6th Workshop on Asian Translation*. Hong Kong: Association for Computational Linguistics, 11 2019.

[60] H. Isozaki, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada, "Automatic evaluation of translation quality for distant language pairs," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2010, pp. 944–952.

[61] R. E. Banchs, L. F. D'Haro, and H. Li, "Adequacy–fluency metrics: Evaluating mt in the continuous space model framework," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 472–482, 2015.

[62] F. Scholz, "Maximum likelihood estimation," *Wiley StatsRef: Statistics Reference Online*, 2014.

[63] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," *arXiv preprint arXiv:1511.06732*, 2015.

[64] S. Shen, Y. Cheng, Z. He, W. He, H. Wu, M. Sun, and Y. Liu, "Minimum risk training for neural machine translation," *arXiv preprint arXiv:1512.02433*, 2015.

[65] D. Bahdanau, P. Brakel, K. Xu, A. Goyal, R. J. Lowe, J. Pineau, A. Memisevic, and Y. Bengio, "An actor-critic algorithm for structured prediction," 2016.

[66] L. Wu, L. Zhao, T. Qin, J. Lai, and T.-Y. Liu, "Sequence prediction with unlabeled data by reward function learning," *IJCAI-17*, pp. 3098–3104, 2017.

[67] J. Kreutzer, A. Sokolov, and S. Riezler, "Bandit structured prediction for neural sequence-to-sequence learning," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2017, pp. 1503–1513.

[68] L. Wu, F. Tian, T. Qin, J. Lai, and T.-Y. Liu, "A study of reinforcement learning for neural machine translation," *arXiv preprint arXiv:1808.08866*, 2018.

[69] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," in *Reinforcement Learning*. Springer, 1992, pp. 5–32.

[70] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning*, 2016, pp. 1928–1937.

[71] M. Cettolo, J. Niehues, S. Stüker, L. Bentivogli, and M. Federico, "Report on the 11th iwslt evaluation campaign, iwslt 2014," in *Proceedings of the International Workshop on Spoken Language Translation, Hanoi, Vietnam*, 2014.

[72] S. Wiseman and A. M. Rush, "Sequence-to-sequence learning as beam-search optimization," *arXiv preprint arXiv:1606.02960*, 2016.

[73] S. Singh, R. Panjwani, A. Kunchukuttan, and P. Bhattacharyya, "Comparing recurrent and convolutional architectures for english-hindi neural machine translation," in *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, 2017, pp. 167–170.

[74] B. Wang, Z. Tan, J. Hu, Y. Chen *et al.*, "Xmu neural machine translation systems for wat 2017," in *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, 2017, pp. 95–98.

[75] C. Isbell, C. R. Shelton, M. Kearns, S. Singh, and P. Stone, "A social reinforcement learning agent," in *Proceedings of the Fifth International Conference on Autonomous Agents*, ser. AGENTS '01. New York, NY, USA: ACM, 2001, pp. 377–384. [Online]. Available: http://doi.acm.org/10.1145/375735.376334

[76] A. C. Tenorio-Gonzalez, E. F. Morales, and L. Villaseñor-Pineda, "Dynamic reward shaping: Training a robot by voice," in *Advances in Artificial Intelligence – IBERAMIA 2010*, A. Kuri-Morales and G. R. Simari, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 483–492.

[77] S. Young, M. Gašić, S. Keizer, F. Mairesse, J. Schatzmann, B. Thomson, and K. Yu, "The hidden information state model: A practical framework for pomdp-based spoken dialogue management," *Computer Speech & Language*, vol. 24, no. 2, pp. 150–174, 2010.

[78] S. R. K. Branavan, D. Silver, and R. Barzilay, "Learning to win by reading manuals in a monte-carlo framework," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 268–277. [Online]. Available: http://dl.acm.org/citation.cfm?id=2002472.2002507

[79] J. Gu, G. Neubig, K. Cho, and V. O. Li, "Learning to translate in real-time with neural machine translation," *arXiv preprint arXiv:1610.00388*, 2016.

[80] K. Nguyen, H. Daumé III, and J. Boyd-Graber, "Reinforcement learning for bandit neural machine translation with simulated human feedback," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 1464–1474.

[81] J. Zhang and C. Zong, "Exploiting source-side monolingual data in neural machine translation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1535–1545.

[82] Y. Cheng, W. Xu, Z. He, W. He, H. Wu, M. Sun, and Y. Liu, "Semi-supervised learning for neural machine translation," *arXiv preprint arXiv:1606.04596*, 2016.