Pivot Based Neural Machine Translation: A Survey

Shivam Mhaskar, Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology Bombay
Mumbai, India
{shivammhaskar, pb}@cse.iitb.ac.in

Abstract

Neural Machine translation models are the best-performing machine translation models. But neural machine translation models are *data hungry*, that is, they require huge amounts of parallel training corpus to produce good results. Such huge amounts of the parallel corpus are not available for many languages, including many Indian languages. In order to overcome this drawback, we use various pivoting techniques. Pivoting techniques refer to using a related pivot language for assisting the source to target translation. Pivoting techniques can be divided into two parts, transfer learning techniques and data augmentation techniques. Multilingual neural machine translation models that can translate between multiple languages and share knowledge between languages can also help tackle the problem of data scarcity. We see that these pivoting techniques are effective in utilizing the resources of the pivot language and give good performance improvements in machine translation models for low-resource languages. In this survey paper, we discuss the various pivoting techniques to utilize the resources of a pivot language to assist the source-target machine translation models.

1 Introduction

The performance of machine translation models has improved rapidly with the help of neural architectures (Dabre et al., 2020). Neural Machine Translation (NMT) models based on the Transformer architecture (Vaswani et al., 2017a) have shown impressive performance but it is only limited to high-resource languages. This is because NMT models are *data hungry*, i.e., they require large amounts of parallel corpus for training. One way to improve the performance of NMT models for low-resource languages is to feed more parallel corpus for training the model. But creating such a parallel corpus is time-consuming and expensive.

Many low-resource languages have a related high-resource language that can be used as an assisting pivot language. The language relatedness between the languages can be utilized to help low-resource languages. Pivot-based transfer learning and data augmentation techniques can be used to utilize the resources of a pivot language to improve the source-target NMT models. In this survey paper, we discuss the work done in pivot-based transfer learning and data augmentation techniques. In pivot-based transfer learning techniques the knowledge representations learned by the source-pivot and pivot-target NMT models are utilized to improve the performance of source-target NMT model. Data augmentation techniques can be used to generate more source-target synthetic data by using the source-pivot and pivot-target parallel corpus.

2 Motivation

India is a land of many languages. In the 8th schedule of the Constitution of India, 23 languages were recognized as official languages. As there are many people in India speaking various languages, it becomes necessary to provide all the information such as healthcare, legal and tourist information in various Indian languages. Educational materials are primarily available in English, but many children across India know only their mother tongue or regional language. The New Education Policy of Government of India advocated to provide educational material in regional languages. Manual translation of different types of content from one language to another has various challenges. Manual translation relies on human bilingual translators who are well versed with both the languages. Manual translation is a time-consuming and costly task. Manual translation of content from domains such as healthcare and education requires translators to have domain knowledge as well. Translation of content with the help of computers, referred to as machine translation, provides a promising solution to challenges of manual translation. Machine trans-

lation is fast, cheap and works across domains.

Neural machine translation models are currently the best performing machine translation models. But a challenge with developing high quality NMT models is that NMT models require a large parallel corpus to provide good results. Therefore, there is a need to come up with techniques to overcome the problem of shortage of parallel corpus. Some techniques are pivoting and multilinguality. In pivoting, a related pivot language comes as an assisting language to help the translation between source to target.

3 Background

3.1 Rule Based Machine Translation

The task of machine translation is done through an analysis-transfer-generation process. This process can be visualized through the Vauquois triangle (Bhattacharyya, 2015). In this process, we first obtain the syntactic representations of the source language sentence. Then we transfer to the target language sentence. Then we convert the syntactic structure to the target language sentence. In rule based machine translation, all the three steps of the process can be performed with predefined rules.

The analysis step makes use of rules of morphological analysis, parsing and semantic generation. These rules are used to obtain the syntactic representation of the input sentence. In the transfer step, bilingual dictionaries are used to translate the words in from the source language to the words in the target language. In the generation step, rules are used to perform syntactic reordering of the words in the target language. This process generates a sentence in the structure of the target language sentences. The drawback with rule based machine translation system is that, the process of rule generation is time-consuming and requires domain knowledge. Also, the rules need to be created separately for each language. Another problem with rule based machine translation systems is that it is unrealistic to cover all rules of the entire process. The handling ambiguity with the rules is also a very tough task. Because of all these drawbacks of the rule based machine translation system, statistical machine translation systems were introduced.

3.2 Statistical Machine Translation

Statistical machine translation involves using statistical methods to tackle the problem of machine translation. Statistical machine translation became popular in the early 2000s. This is because during that time, the performance and adoption of computers were improving at a rapid pace. With this rapid adoption of computer, the amount of digitally available text corpus also increased. This digitally available text corpus could be used to implement statistical methods using the computers. In statistical machine translation systems, the task of translation is modelled by a conditional probability distribution of finding the target sentence given the source sentence. This can be further split into two different models, the language model and a translation model. The language model assigns a probability to a sentence, which represents how probable is that the sentence belongs to the given language. The translation model gives a probability that a word in source language translates to a word in target language. The translation model further models fertility and distortion. These language model and translation models are given to the decoder. The decoder of the statistical machine translation system produces the target language sentence based on the probability values obtained from language model and translation model and performing a search over the derived hypothesis.

Statistical machine translation models were able to improve the performance over the rule based systems by leveraging the statistical methods. But the statistical machine translation systems are limited by the available training parallel corpus. Statistical machine translation systems require good amount of parallel corpus to produce good translation model and language model. Also, a good amount of parallel corpus is required to model as many words of a language as possible, or else many words in the language will be unseen by the model. Statistical machine translation models work on word translation probabilities, but for many language pairs, translation happens between phrases. In order to overcome this drawback, phrase based statistical machine translation models were introduced.

3.3 Neural Machine Translation

Deep neural networks were able to solve a lot of problems with very good performance. With the growing power of deep neural networks, they were used to tackle the task of machine translation. As the task of machine translation is a sequence to sequence learning task where the length of the input sequence and output sequence can vary, a variant

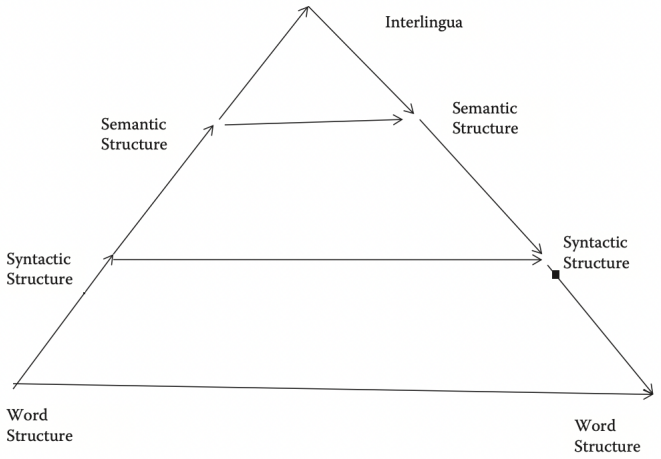

Figure 1: Vauquois triangle

of deep neural networks called recurrent neural networks were used to implement neural machine translation models. Recurrent neural networks are able to model the sequential nature of the data. Recurrent neural network based machine translation models are encoder decoder based models, where an encoder encodes the source sentence and the decoder decodes the output. Recurrent neural networks suffered from the problem of squeezing all the context of the input sentence into a single fixed length vector.

This bottleneck problem of recurrent neural networks was solved with the help of an attention mechanism. In the attention mechanism, the decoder performs a soft search over the input sequence or attends the input sequence while producing output at each time step. The decoder gives attention to the proper subset of input sequence required to produce the output at the current time step. Recurrent neural networks with attention mechanism improve the performance over normal recurrent neural networks. Transformer (Vaswani et al., 2017b) architecture introduced for neural machine translation was based on only attention mechanism. Transformer overcame the sequential processing nature of the recurrent neural networks and achieved state-of-the-art results for neural machine translation for various language pairs.

3.4 Pivoting

Pivoting refers to the set of techniques in which a pivot language is used to assist the task of translation between source to target language. The resources of the pivot language are used to improve the performance of source to target machine translation model. The pivoting techniques are based on transfer learning and data augmentation. In transfer learning, the representations learned in the source to pivot model and pivot to target model are used to improve the performance of source to target. These models have already learned some language knowledge, and this knowledge can be utilized by the source to target model. In the data augmentation based techniques, the pivot to source and pivot to target models are used to convert the pivot to target and pivot to source parallel data. This gives source to target augmented parallel data. This augmented parallel corpus is added to the original parallel corpus, and the model is trained on this combined dataset. This augmented parallel corpus provides additional data for training the machine translation model.

3.5 Multilingual Neural Machine Translation

In a Multilingual neural machine translation model, a single model is trained to translate between multiple language pairs. Multilingual neural machine translation models have the advantage of being easy to train and deploy. And as a single multilingual neural machine translation model can translate between multiple language pairs, there is no need for training and storing multiple bilingual machine translation models. And as the parameters between all the languages are shared, the knowledge learned while training for one language pair helps the task of translation between other language pairs. This especially helps low resource language pairs which do not have much parallel data. As the representations learned from the data of high resource languages is shared for the low resource languages as

well.

3.6 Machine Translation Evaluation

3.6.1 Human Evaluation

Human Evaluation of machine translation output is performed by a human evaluator who has knowledge of both the source and target language. The human evaluator gives a score to each translated output based on a predefined factor. The scoring methodology can have a single score or multiple scores based on predefined factors. Human evaluation is a costly and time-consuming process. Human Evaluation also requires recruiting human evaluators who have expertise in both the source and target language involved. However, human evaluation provides a high quality evaluation of the machine translation output. The popular scoring methodology to score machine translation output consists of scoring based on Adequacy and Fluency. Adequacy The adequacy of a translation refers to how well the information or meaning in the source sentence is translated into the target sentence. Adequacy is an important metric for evaluating translations because it is essential that the proper meaning of the source sentence appears in the target sentence. An adequacy score is given manually by an evaluator based on if the meaning of the source sentence is translated to the target sentence properly or not. Fluency The fluency of a sentence refers to how well-formed the sentence is in that particular language. A highly fluent sentence is one which will be produced by a native speaker of the language. The fluency of a translated sentence is scored by a human evaluator by looking only at the target sentence. Fluency of a sentence depends on the choice of words and the word order in the sentence.

3.6.2 Automatic Evaluation

BLEU BLEU which stands for Bilingual Evaluation Understudy (Papineni et al., 2002) is an evaluation metric used to evaluate the translations produced by machine translation systems. A BLEU score is computed between the hypothesis sentences generated by the machine translation system and the reference sentence generated by a human translator. The BLEU score measures the closeness between the hypothesis and the reference sentence. The BLEU score makes use of modified n-gram precision. In modified n-gram precision, we clip the count of a candidate word by the count of the word in the reference sentence. The BLEU score is computed by taking the weighted sum of the n-gram precision. BLEU score also has a brevity penalty factor which is used to penalize the longer length hypothesis sentences.

$$BP = \{\, 1 \; if c > re^{(1-r/c)} if c \leq r \qquad (1)$$

$$BLEU = BP.exp(\sum_{n=1}^{N} w_n log p_n) \qquad (2)$$

Subword BLEU Subword BLEU is a variant of BLEU score in which the BLEU score is computed on sentences of subwords rather than sentences of words. Subword BLEU is especially useful for evaluating morphologically rich languages like Indian languages. Subword BLEU is computed by first converting the words in hypothesis and reference sentences by using subwordization techniques like BPE. Then BLEU score is computed on the sentences of subwords.

4 Dataset

4.1 Samanantar Dataset

The Samanantar (Ramesh et al., 2022) corpus is a parallel corpus dataset for Indian languages. The Samanantar corpus contains parallel corpus between English and 10 Indian languages. The Samanantar corpus has 2 parts. The first part is combination of all the parallel corpora available on the internet for all the language pairs. The second part consists of a parallel corpus obtained by picking parallel sentences from a comparable corpus.

4.2 Anuvaad Parallel Corpus

The Anuvaad Parallel corpus is developed as a part of the Anuvaad Project. The Anuvaad parallel corpus contains parallel corpus between English and 11 Indian languages. The Anuvaad parallel corpus consists of sentences from various domains. The Anuvaad parallel corpus for English-Marathi contains 23 lac parallel sentences.

4.3 Low Resource MT Workshop Dataset

The Low Resource MT Workshop (Ojha et al., 2020) conducts shared tasks based on machine translation tasks for low resource language pairs. The 2021 edition of the low resource MT workshop had an English-Marathi task. The task provided an English-Marathi parallel corpus on the Covid-19 domain. The parallel corpus consists of 18,000 English-Marathi parallel sentences.

	as	bn	gu	hi	kn	ml	mr	or	pa	ta	te	Total
as	–	356	142	162	193	227	162	70	108	214	206	1839
bn		–	1576	2627	2137	2876	1847	592	1126	2432	2350	17920
gu			–	2465	2053	2349	1757	529	1135	2054	2302	16361
hi				–	2148	2747	2086	659	1637	2501	2434	19466
kn					–	2869	1819	533	1123	2498	2796	18168
ml						–	1827	558	1122	2584	2671	19829
mr							–	581	1076	2113	2225	15493
or								–	507	1076	1114	6218
pa									–	1747	1756	11336
ta										–	2599	19816
te											–	20453

Figure 2: Number of parallel sentences in Samanantar parallel corpus

4.4 OPUS parallel corpus

For low resource languages such as Nepali, Konkani and Sinhala we used the OPUS (Tiedemann, 2012) corpora for training the NMT models. We used parallel corpora such as GNOME, KDE, Ubuntu, Ted (Reimers and Gurevych, 2020) and Bible from the Opus website.

Language Pair	Number of Sentence Pairs
English-Nepali	151K
English-Hindi	8.5M
Hindi-Nepali	379K
English-Konkani	46K
English-Marathi	3.3M
Marathi-Konkani	46K
Hindi-Konkani	46K
Hindi-Marathi	1.9M
Sinhala-English	641K
Sinhala-Tamil	363K
Tamil-English	5.1M

Table 1: Dataset Statistics

5 Pivoting in Statistical Machine Translation

A naive approach to pivoting is to translate the source sentence to a pivot sentence using a source-pivot machine translation system and then pass this pivot sentence through a pivot-target machine translation system to generate the final target language sentence (De Gispert and Marino, 2006). Utiyama and Isahara (2007) improved this approach by generating multiple pivot sentences which are then separately passed through the pivot-target machine translation system to produce multiple target sentences and the target sentence with the highest score is picked as the final translation. The use of pivoting techniques were initially done in phrase based statistical machine translation systems (Dabre et al., 2015). Phrase based statistical machine translation systems make use of phrase tables, which consists of phrase translation probabilities. These phrase tables are generated from parallel corpus. If the amount of parallel corpus is small, then the phrase tables will not be of high quality, and they also may not contain many phrase pairs. In such cases a pivot language can be used to generate source-pivot and pivot-target phrase tables and then phrase table triangulation can be performed to create a pivot based phrase table. Then we can combine the direct phrase table and pivot based phrase tables using various techniques. The phrase tables generated using this technique also contains phrase pairs which are not present in the direct phrase table. Also, the phrase table generated using this technique gives good performance improvement compared to the direct phrase table.

The source to pivot and pivot to target phrase tables are generated using the source-pivot and pivot-target parallel data. The phrase table contains 4 values, the forward and inverse phrase translation probabilities and the forward and inverse lexical translation probabilities. In order to obtain the source-target pivot based phrase table, phrase table triangulation is performed over the source-pivot and pivot-target phrase tables to obtain the values.

$$\theta(f|e) = \sum_{p_i} \theta(f|p_i) * \theta(p_i|e) \quad (3)$$

$$\theta(e|f) = \sum_{p_i} \theta(e|p_i) * \theta(p_i|f) \quad (4)$$

$$P_w(f|e,a) = \sum_{p_i} P_w(f|p_i,a_1) * P_w(p_i|e,a_2) \quad (5)$$

$$P_w(e|f,a) = \sum_{p_i} P_w(e|p_i,a_2) * P_w(p_i|f,a_1) \quad (6)$$

Once the direct phrase table and the pivot based phrase tables are obtained, then the next step is to combine these phrase tables. The phrase tables can be combined using various techniques. Some techniques are linear interpolation, fillup interpolation and multiple decoding paths. In linear interpolation technique, the probability values of the phrases in the source-target phrase table is obtained by taking a weighted sum of the values from direct phrase table and the pivot based phrase table.

$$\theta(f|e) = \alpha_0 * \theta_d irect(f|e) + \sum_{l_i} \alpha_{l_i} * \theta_{l_i}(f|e) \quad (7)$$

In the fillup interpolation technique, the probability values are not modified. But we add new phrase pairs from the pivot based phrase table if the phrase pair is not already present in the direct phrase table. Because of this we get additional phrase pairs we could not be obtained from direct phrase tables. In the multiple decoding paths technique, the phrase tables are not combined beforehand. But during the decoding time, multiple phrase tables are used to perform the decoding. In this technique, the phrase tables are kept separate.

6 Pivoting in Neural Machine Translation

Zoph et al. (2016) first trained a parent model on a high-resource language which is then used to initialize the parameters of a child model which is finetuned on a low-resource language. Kim et al. (2019) proposed pivot-language-based transfer learning techniques for NMT in which the encoder and decoder of source-pivot and pivot-target NMT models are used to initialize the source-target model. Ko et al. (2021a) exploited the linguistic overlap between related languages to adapt NMT models of high-resource languages for low-resource languages through techniques like denoising autoencoding, back-translation, and adversarial objectives.

6.1 Pivot based Transfer Learning for Neural Machine Translation

In machine learning, transfer learning refers to utilizing the knowledge gained for performing one task for some other task. This is done by using the machine learning model trained to perform one task as initialization for performing some other task. In neural machine translation, transfer learning can be performed by using the source to pivot and pivot to target models. The parameters of these models can be used to initialize the source to target models in various ways. Also, the process in which these models are trained, and the parameters are initialized can also be performed in various ways.

One way to initialize the parameters of the source to target model is to initialize the encoder of the source to target model with the encoder of the source to pivot model and the decoder of the source to target model with the decoder of the pivot to target model (Kim et al., 2019). This type of initialization is performed because the encoder of the source to pivot model has learned representations or knowledge for the source language, and that is why it can be used to initialize the encoder of the source to target model. Similarly, the decoder of the pivot to target model has learned the representations for the target model and can be used to initialize the decoder of the source to target model. Once the encoder and decoder of the source to target model are initialized, the model is trained on source-target parallel data.

A problem with the first approach is that the encoder in the source to pivot model is trained to produce outputs for the pivot decoder and not the target decoder. And the decoder of the pivot to target model is trained on the outputs of the pivot encoder and not the source encoder. In order to overcome this drawback, a step wise pretraining strategy is followed to train the models. In the first step, a source to pivot model is trained on source-pivot parallel data. In the next step, the encoder of the source to pivot model is used to initialize the encoder of the pivot to target model. Now the pivot to target model is trained on the pivot to target data, but the encoder is frozen. This means that the parameters of the encoder are not updated. This retains the source language representations in the encoder learned in the first step. This also prevents the encoder from adapting to the pivot language. Now the encoder is producing representations from the source encoder which is used by the target decoder. In this way, the drawback of first transfer learning approach is mitigated. In the next step, the encoder and decoder of the model from the second step is used to initialize the encoder and decoder of

Pivot Language	Sentence Strategy	Standalone	Linear Interpolate (1) With Direct	Linear Interpolate (2) With Direct	Fill Interpolate With Direct	MDP With Direct
1. Direct	**33.86**					
2. Chinese	23.53	28.89	34.03	**34.61**	34.31	**35.66**
3. Korean	26.30	28.92	**34.65**	34.18	34.64	**35.60**
4. Esperanto	22.43	28.73	**34.63**	**34.55**	35.32	**35.74**
5. Paite	19.40	26.64	34.17	34.40	**34.66**	**35.22**
6. Marathi	15.68	21.80	33.88	33.80	33.83	34.03
7. Kannada	16.94	24.15	33.74	34.13	**34.87**	**35.52**
8. Telugu	14.15	21.31	33.81	33.85	34.04	**34.57**

Figure 3: Pivoting in SMT results: Japanese-Hindi

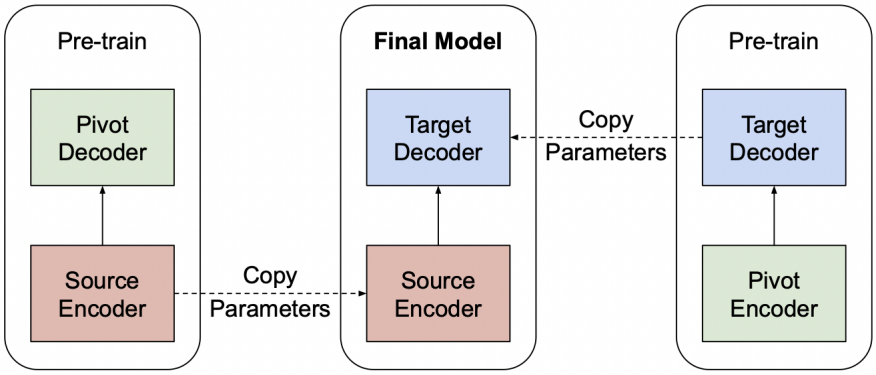

Figure 4: Plain Transfer Learning

source to target model. The source to target model is then trained on source to target parallel data.

6.2 Adapting High Resource NMT models for Low Resource Languages

Many low resource languages are related to a high resource language. For Indian languages, many are related to Hindi. Hindi is a relatively high resource language. A neural machine translation system for low resource languages can be created by adapting a high resource language neural machine translation model (Ko et al., 2021b). There are various techniques in which a high resource language model can be adapted to a related low resource language.

We can consider the task of translating from English to low resource language. In the first step, we train an English to high resource language pair. The second step is called as denoising auto encoding. In this step we noise the sentence and the task of the model is to predict the original sentence. The noising can be performed by shuffling the words in a sentence in such a way that no word is shuffled 3 positions from its original position and masking the words in a sentence with a certain probability. The denoising auto encoding is performed for high resource and low resource language sentences.

The next step is backtranslation to generate augmented data. In backtranslation we train a reverse model from language B to language A, and then we translate the monolingual data of language B to language A using this model. After doing this, we obtain the augmented parallel data between language A and language B. In this step, backtranslation is performed in an iterative manner to start from a high resource language to English model. The monolingual low resource language data is passed to the high resource language to English model and translated to English. After this, we get the low resource language to English augmented data. We use this low resource language to English data to train the reverse high resource language to English model further. In the next iteration, we again translate the monolingual low resource language data to English. But in the second iteration, the model is trained on some low resource language to English augmented data. Then we again train the model further on this new data. This process

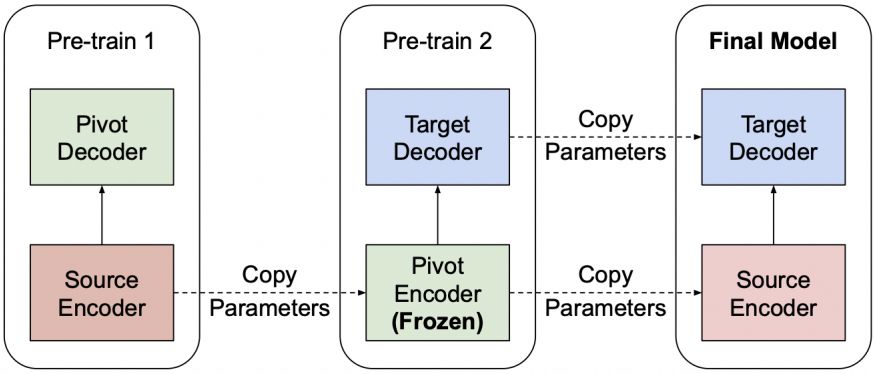

Figure 5: Stepwise Transfer Learning

	French→German				German→Czech			
	newstest2012		newstest2013		newstest2012		newstest2013	
	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]	BLEU [%]	TER [%]
Direct source→target	14.8	75.1	16.0	75.1	11.1	81.1	12.8	77.7
Multilingual many-to-many	18.7	71.9	19.5	72.6	14.9	76.6	16.5	73.2
Multilingual many-to-one	18.3	71.7	19.2	71.5	13.1	79.6	14.6	75.8
Plain transfer	17.5	72.3	18.7	71.8	15.4	75.4	18.0	70.9
+ Pivot adapter	18.0	71.9	19.1	71.1	15.9	75.0	18.7	70.3
+ Cross-lingual encoder	17.4	72.1	18.9	71.8	15.0	75.9	17.6	71.4
+ Pivot adapter	17.8	72.3	19.1	71.5	15.6	75.3	18.1	70.8
Step-wise pre-training	18.6	70.7	19.9	70.4	15.6	75.0	18.1	70.9
+ Cross-lingual encoder	**19.5**	**69.8**	**20.7**	**69.4**	**16.2**	**74.6**	**19.1**	**69.9**

Figure 6: Pivot based Transfer Learning Results: Japanese-Hindi

can be performed for k iterations.

In the next step, we try to make the representations of the model language agnostic. This needs to be done so that the model produces language agnostic representations. This helps the model adapt from high resource language to low resource language, as the encoder output will not have any language specific information. In order to achieve this, the encoder is trained using a discriminator. We use two discriminators for this step. The first discriminator is trained to discriminate between low resource and high resource language. The second discriminator is trained to discriminate between English and all other languages.

6.3 Transformer Cross Attention Fine Tuning

A simple way of applying transfer learning in neural machine translation is to first train a source to pivot neural machine translation model. Now the target-side of the model is changed from pivot language to target language. But the problem with this approach is that when the model is trained on source-target data, the knowledge learned during the source-pivot translation task is overwritten or forgotten. This is not ideal because the representations learned from the source to pivot machine translation model which is a high resource model can help the source to target model. In order to overcome this problem, we can freeze some parameters of the model while training on the source-target parallel data.

One of the study analyzes the importance of the cross attention layer in the Transformer architecture and studies how parameter efficient finetuning can be performed (Gheini et al., 2021). In the first experiment, the source to pivot model is used to initialize the source to target model. Then all the parameters of the model are frozen except for the source and target embeddings. This experiment is performed to see the performance of the model after finetuning only the embeddings layer against training the entire model from scratch. In the second experiment, the source to target model is initialized from the source to pivot model and all the param-

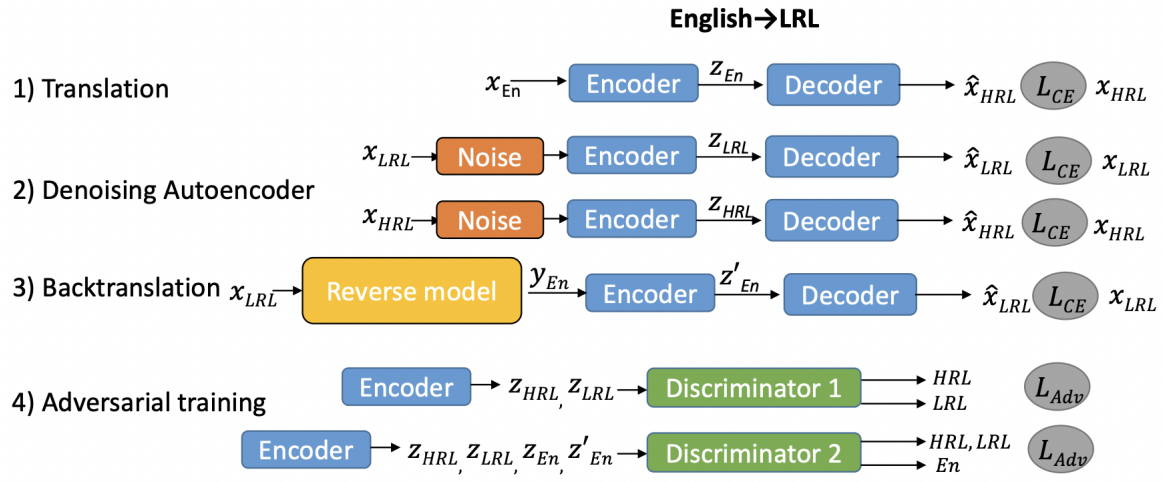

Figure 7: Adapting English to HRL model for English to LRL

En→ LRL		**Un-adapted Model**	**Adapted Models**			
LRL	**HRL**	En→HRL	Adv	BT	BT+Adv	BT+Adv+fine-tune
Portuguese	Spanish	3.8	10.1	14.8	18.0	**21.2**
Catalan	Spanish	6.8	9.1	21.2	22.5	**23.6**
Marathi	Hindi	7.3	8.4	9.5	15.6	**16.1**
Nepali	Hindi	11.2	17.6	16.7	25.3	**26.3**
Urdu	Hindi	0.3	3.4	0.2	**7.2**	-
Egyptian Arabic	MSA	3.5	3.8	**8.0**	**8.0**	**8.0**
Levantine Arabic	MSA	2.1	2.1	4.8	**5.1**	4.7

Figure 8: Results of Adapting HRL models to LRL models

eters of the model are frozen except for the cross attention layer and embeddings layer. This experiment is performed to the performance of finetuning the cross attention layer against finetuning only the embeddings layer and training the entire model from scratch. In the third experiment, the source to pivot model is used to initialize the source to target model, but the cross attention layer is not initialized. While finetuning only the cross attention layer and embeddings layer is trained and rest of the layers are frozen. This experiment is performed to check the importance of using pretrained cross attention layer from the source to pivot model.

The results of the experiment show that, finetuning the embeddings layer, the cross attention layer and finetuning the entire model improves performance over training the model from scratch. Finetuning the cross attention layer along with the embeddings layer improves the performance over just finetuning the embeddings layer. Also, the performance of finetuning the cross attention layer comes close to finetuning the entire model. Finetuning the randomly initialized cross attention layer gives poor performance as compared to finetuning the initialized cross attention layer. This experiment shows the importance of the knowledge learned in the cross attention layer during the source to pivot task while finetuning for the source to target task.

6.4 Data Augmentation

Sennrich et al. (2016) proposed the backtranslation technique in which the synthetic data is created by translating monolingual data. Sen et al. (2021) proposed the phrase pair injection technique in which source-target phrase pairs generated from the source-target parallel corpus using SMT are augmented with source-target parallel corpus. The bad-quality phrase pairs can be filtered out using Labse-based (Feng et al., 2022) filtering techniques (Batheja and Bhattacharyya, 2022). The pivot sentences of the source-pivot parallel data can be translated to the target language using a pivot-target NMT model to generate synthetic source-target parallel corpus (Xia et al., 2019).

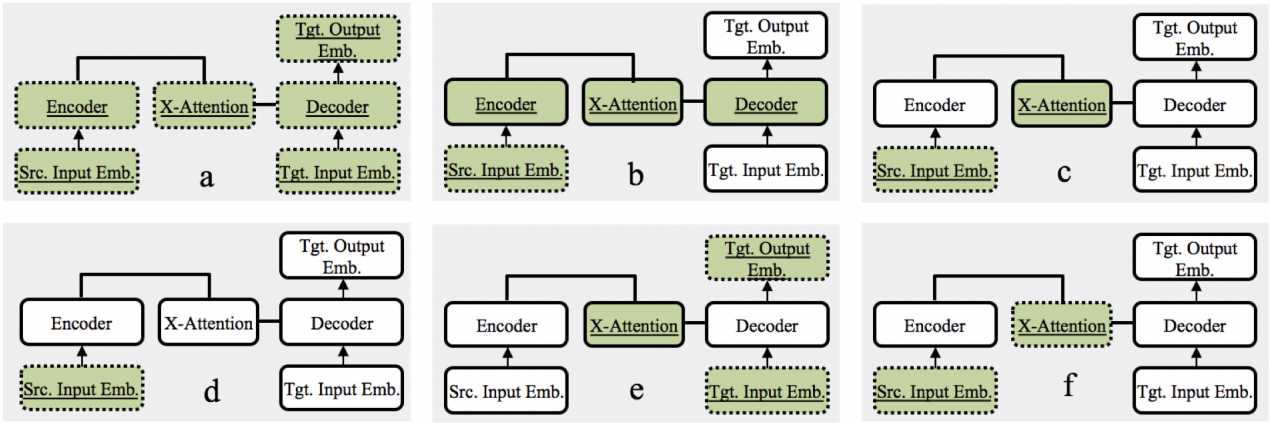

Figure 9: Cross attention finetuning

	Ro–En	Ja–En	De–En	Ha–En	Fr–Es	Fr–De
`scratch` (100%)	29.0	9.2	30.8	5.4	24.4	18.5
`{src,tgt}` (8%)	29.8	8.7	32.4	**8.6**	21.6	11.6
`{src,tgt}+body` (75%)	**31.0**	**11.8**	**36.2**	**8.8**	**27.3**	**21.4**
`{src,tgt}+xattn` (17%)	**(-0.1) 30.9**	**(-2.0) 9.8**	**(-1.2) 35.0**	(-0.4) 8.4	**(-0.8) 26.5**	**(-1.8) 19.6**
`{src,tgt}+randxattn` (17%)	27.9	8.4	33.3	7.0	26.0	18.8

Figure 10: Results of Cross Attention Finetuning

In pivot language based data augmentation techniques, the parallel corpus containing the pivot language is converted to source or target language. In the first step, we train a pivot to target and a pivot to source model using the pivot-target and pivot-source parallel corpus. Then the pivot side of the pivot-source parallel corpus is translated to the target language using the pivot to target model. And the pivot side of the pivot-target parallel corpus is translated to source language using the pivot to source model. This generates source-target augmented parallel corpus. This augmented corpus is added to the training corpus and the model is trained on this combined dataset.

The advantage with this technique is that as the pivot language is a high resource language, the pivot to source and pivot to target are high quality machine translation system. Hence, the translated data is of decent quality. This decent quality data is suitable for training a machine translation system.

7 Multilingual Neural Machine Translation

Multilingual Neural Machine Translation models (Johnson et al., 2017) are models that can translate from a single language to multiple languages, multiple languages to a single language or multiple languages to multiple languages. Multilingual neural machine translation models can be implemented in various ways with the encoder-decoder based Transformer architecture.

7.1 One to Many

One to Many multilingual neural machine translation model is a model that can translate from one source language to multiple target languages. One to Many multilingual neural machine translation model can be implemented in multiple ways. One way is to have a single encoder for the source language and multiple decoders for each target side language. The disadvantage with this architecture is that there is no sharing of knowledge between the target side languages. In order to overcome this drawback, we can have a single encoder and a single shared decoder for all target side languages. As all the target side languages are sharing all the decoder parameters, there is sharing of knowledge between all the target side languages.

One of the challenges with implementing a multilingual neural machine translation model with a single encoder and a single shared decoder is that when training the model on a parallel corpus, then the model does not know to which language should the source language sentence be translated. Dur-

ing inference, we also need a method to specify the model to produce a particular required target language sentence. One way to do this is to make of language tokens. Language tokens are special tokens corresponding to each language added to the vocabulary. During training, the source language sentences are prepended with the language token of the target side language. This specifies the model, the language of target side reference sentence. During inference time, the input source sentence is prepended with the token of the required target language. Then the model produces the output translation in the corresponding target language. In this way, we can specify the target language to which the model should translate to during training and inference.

7.2 Many to One

Many to one neural machine translation model is a neural machine translation model that can translate from multiple languages to a single language. One way to implement a many to one multilingual neural machine translation model is to have a separate encoder for each source side language and a single decoder. But the disadvantage with this architecture is that there is no sharing of knowledge on the source side. The encoders of the model are trained in isolation. In order to overcome this disadvantage, we can have a single shared encoder for all source languages and a single decoder for the target side language. The advantage with this architecture is that now the source languages share all the parameters between them. This enables sharing of knowledge between all the source side languages.

In one to many multilingual neural machine translation model, a language token was used to specify to which target language the model should translate. This was done to specify the model to which target language the reference sentence belongs. But in many to one multilingual neural machine translation model there is only a single target side language, so there is no need to specify to which language the model should translate. The model translates the source language to the single target language. A source side language token can also be prepended to the source language to specify to the encoder which source language the input sentence belongs.

7.3 Many to Many

Many to many neural machine translation model is a neural machine translation model that can translate from multiple source languages to multiple target languages. If we want to use bilingual translation models to translate from 'n' languages to 'n' languages, then we will require '$n*(n-1)$'. This is a huge number of models and grows quadratically with the increase in number of languages. A single multilingual neural machine translation model can replace these '$n * (n - 1)$' with a single model. This reduces the number of neural machine translation models to be trained drastically. This also provides huge benefits to training and deploying the machine translation model.

A multilingual neural machine translation model can be implemented by using a single shared encoder for all source languages and a single shared decoder for all target languages. This sharing of parameters enables sharing of knowledge on the encoder side as well as decoder side. So, the representations learned for one language are utilized for all other languages. This sharing of knowledge especially helps low resource language pairs. As low resource language pairs have low amount of parallel data, the model is not able to learn much from the limited available data. In multilingual neural machine translation model, the representations learned while training for high resource language pair can be utilized while training for the low resource language pair. In this way, multilingual neural machine translation model improves the performance for low resource language pairs.

In many to many multilingual neural machine translation, we have multiple languages at the source side and multiple languages at the target side. So during training, we need to specify to the model to which target language the reference sentence belongs. And at inference time, we need to specify to the model to which target language the source language sentence should be translated. This can be done with language tokens. In many to many multilingual neural machine translation models, language tokens can be used in two ways. In the first technique, we prepend each source language sentence with the language token of the target language. And during inference, the language token of the target language to which the source sentence should be translated to is prepended to the source language sentence. This specifies the model to which target language the source language sentence should be translated. In the second technique, we prepend the source language token to all the source language sentences. This specifies the model to

which source language the sentence belongs. And during decoding the target sentence, the first token given to the decoder is the language token of the target language sentence. So after the model has encoded the source language sentence when the model starts decoding, the first token given to the decoder that is the language token specifies to the model to which target language the sentence should be translated. Then the model starts decoding in the corresponding target language. In this way, many to many multilingual neural machine translation model can be implemented.

8 Summary

In this survey paper, we first discussed the various paradigms of machine translation. Then we discussed in detail the statistical machine translation systems. Then we discussed the various neural machine translation systems. Then we discussed in detail the various pivoting techniques in machine translation. We first looked at how pivoting can be applied in statistical machine translation systems. Then we discussed in detail the various pivoting techniques in neural machine translation systems. Finally, we discussed in detail the multilingual neural machine translation systems.

9 Conclusion

In this survey paper, we looked at how the neural machine translation models give the best performance for machine translation models. We saw that neural machine translation models require a huge amount of parallel corpus to train good neural machine translation systems. We saw that not many languages have such huge amounts of parallel corpus. Thus, there is a need to use pivoting techniques to utilize the resources of high-resource language pairs for translation between low-resource language pairs. We looked at how pivoting techniques can help improve the performance of machine translation systems for low-resource languages. We can also conclude that multilingual neural machine translation models can also utilize the knowledge learned for one language pair for other language pairs.

10 Future Work

The short term research direction is to use multiple pivot languages for assisting the task of translation from source to target languages. The resources of these multiple pivot languages can be used while implementing the source-to-target machine translation systems. The midterm research direction is to combine multiple pivoting techniques. Multiple experiments can be performed combining multiple pivoting techniques and finding the best combination of techniques that give the best performing model. The long term research direction is to tackle the problem of catastrophic forgetting in transfer learning. This problem can be tackled by making changes to the underlying neural machine translation models and modifying the learning strategies.

References

Akshay Batheja and Pushpak Bhattacharyya. 2022. Improving machine translation with phrase pair injection and corpus filtering.

Pushpak Bhattacharyya. 2015. *Machine translation*. CRC Press.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A survey of multilingual neural machine translation. *ACM Computing Surveys (CSUR)*, 53(5):1–38.

Raj Dabre, Fabien Cromieres, Sadao Kurohashi, and Pushpak Bhattacharyya. 2015. Leveraging small multilingual corpora for SMT using many pivot languages. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1192–1202, Denver, Colorado. Association for Computational Linguistics.

Adrià De Gispert and Jose B Marino. 2006. Catalan-english statistical machine translation without parallel corpus: bridging through spanish. In *Proc. of 5th International Conference on Language Resources and Evaluation (LREC)*, pages 65–68.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Mozhdeh Gheini, Xiang Ren, and Jonathan May. 2021. Cross-attention is all you need: Adapting pretrained Transformers for machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1754–1765, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's

multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Yunsu Kim, Petre Petrov, Pavel Petrushkov, Shahram Khadivi, and Hermann Ney. 2019. Pivot-based transfer learning for neural machine translation between non-English languages. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 866–876, Hong Kong, China. Association for Computational Linguistics.

Wei-Jen Ko, Ahmed El-Kishky, Adithya Renduchintala, Vishrav Chaudhary, Naman Goyal, Francisco Guzmán, Pascale Fung, Philipp Koehn, and Mona Diab. 2021a. Adapting high-resource NMT models to translate low-resource related languages without parallel data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 802–812, Online. Association for Computational Linguistics.

Wei-Jen Ko, Ahmed El-Kishky, Adithya Renduchintala, Vishrav Chaudhary, Naman Goyal, Francisco Guzmán, Pascale Fung, Philipp Koehn, and Mona Diab. 2021b. Adapting high-resource nmt models to translate low-resource related languages without parallel data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 802–812.

Atul Kr Ojha, Valentin Malykh, Alina Karakanta, and Chao-Hong Liu. 2020. Findings of the loresmt 2020 shared task on zero-shot for low-resource languages. In *Proceedings of the 3rd Workshop on Technologies for MT of Low Resource Languages*, pages 33–37.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Divyanshu Kakwani, Navneet Kumar, et al. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Sukanta Sen, Mohammed Hasanuzzaman, Asif Ekbal, Pushpak Bhattacharyya, and Andy Way. 2021. Neural machine translation of low-resource languages using smt phrase pair injection. *Natural Language Engineering*, 27(3):271–292.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Masao Utiyama and Hitoshi Isahara. 2007. A comparison of pivot methods for phrase-based statistical machine translation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 484–491, Rochester, New York. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017a. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017b. Attention is all you need. *Advances in neural information processing systems*, 30.

Mengzhou Xia, Xiang Kong, Antonios Anastasopoulos, and Graham Neubig. 2019. Generalized data augmentation for low-resource translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5786–5796, Florence, Italy. Association for Computational Linguistics.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.