EduMT: Developing Machine Translation System for Educational Content in Indian Languages

Ramakrishna Appicharla, Asif Ekbal, Pushpak Bhattacharyya
Department of Computer Science and Engineering
Indian Institute of Technology Patna
Patna, Bihar, India
{appicharla_2021cs01,asif,pb}@iitp.ac.in

Abstract

In this paper, we explore various approaches to build Hindi to Bengali Neural Machine Translation (NMT) systems for the educational domain. Translation of educational content poses several challenges, such as unavailability of gold standard data for model building, extensive uses of domain-specific terms, as well as the presence of noise in the form of spontaneous speech as the corpus is prepared from subtitle data and noise due to the process of corpus creation through back-translation. We create an educational parallel corpus by crawling lecture subtitles and translating them into Hindi and Bengali using Google translate. We also create a clean parallel corpus by post-editing synthetic corpus via annotation and crowd-sourcing. We build NMT systems on the prepared corpus with domain adaptation objectives. We also explore data augmentation methods by automatically cleaning synthetic corpus and using it to further train the models. We experiment with combining domain adaptation objective with multilingual NMT. We report BLEU and TER scores of all the models on a manually created Hindi-Bengali educational testset. Our experiments show that the multilingual domain adaptation model outperforms all the other models by achieving 34.8 BLEU and 0.466 TER scores.

1 Introduction

Massive Open Online Courses (MOOCs) have gained a lot of attention in recent years due to the availability of high-quality educational resources free of cost. In India, National Programme on Technology Enhanced Learning (NPTEL)[1] is one such initiative to promote online education. However, most of the content offered in English poses a problem for non-native English language speakers especially in a multilingual country like India. One potential solution to mitigate this problem is developing Machine Translation (MT) systems to translate contents from English to other Indian languages. Developing Machine Translation (MT) systems between two Indian languages is more difficult than developing systems between English and Indian languages due to the unavailability of the educational parallel corpus for Indian languages. MT systems, especially current state-of-the-art Neural Machine Translation (NMT) systems (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017) are data-hungry and requires a lot of training data (Zoph et al., 2016; Koehn and Knowles, 2017). Developing MT systems for the educational domain poses issues such as lack of data, translation of domain-specific terms, phrases, and mathematical expressions. Since the dataset is prepared from the lecture subtitles and transcripts, it also contains noise in the form of spontaneous speech (e.g: 'umm', 'yes! good morning' etc.) and repetition of phrases (e.g: 'ok, well.. ok well, now we have to compute this value' etc.). Due to these issues, building an MT system for the educational domain is a challenging task.

In this paper, we focus on developing the NMT systems between two Indian languages, namely Hindi → Bengali language pair for the computer science domain. We create a Hindi-Bengali educational corpus by crawling NPTEL lecture subtitles, transcripts that are in English and translating them into Hindi and Bengali. We create two types of educational parallel corpora, *'synthetic'* and *'clean'*. *Synthetic* corpus is prepared from translating English data into Hindi and Bengali with the help of Google Translate[2]. *Clean* corpus is prepared by manual post-editing of synthetic data via manual annotation and crowd-sourcing. We conduct experiments on prepared corpora with domain adapta-

[1] https://nptel.ac.in/

[2] https://translate.google.com/

Proceedings of the 18th International Conference on Natural Language Processing, pages 35–43
Silchar, India. December 16 - 19, 2021. ©2021 NLP Association of India (NLPAI)

tion (Chu et al., 2017), objective. We experiment with denoising (Edunov et al., 2018) and automatic post-editing (Pal et al., 2016) objectives to automatically clean the *synthetic* corpus which are further used to train the models. We also experimented on multilingual NMT (Johnson et al., 2017) using the English part of the corpus along with Hindi and Bengali. Since there is no standard educational test set is available to test our models' performance, we manually create the test set by translating the Hindi part of the English-Hindi parallel corpus (AI domain) from Adap-MT shared task (Sharma et al., 2020) into Bengali. We report BLEU and TER score (Post, 2018) on the prepared test corpus[3].

The paper is organized as follows. In Section 2, we briefly review a few of the notable works on building MT systems for educational content and domain adaptation, data augmentation methods in NMT. In Section 3, we describe the corpus creation process. The experimental setup used to conduct domain adaptation and semi-synthetic data augmentation experiments are described in Section 4 and Section 5, respectively. The NMT model settings and experimental setup are described in Section 6. Results are described in Section 7. Finally, the work is concluded in Section 8.

2 Related Work

Building an NMT system for any domain requires a significant amount of data. In the educational domain, obtaining data is very challenging. Most of the works in building MT systems for the educational domain is focused on creating corpora. Abdelali et al. (2014) have created educational corpora for 20 languages (20 monolingual and 190 parallel corpora) by crawling AMARA website[4] which is a community-driven web-based platform for editing and creating subtitles for videos. Parallel corpus for European languages in Educational domain have been created via crowd-sourcing Kordoni et al. (2016); Sosoni et al. (2018); Behnke et al. (2018) . They built NMT systems on the prepared corpora and report that even a small amount of crowd-sourced translations can improve the translation quality.

Domain adaptation is a methodology to adapt models trained on out-of-domain data to in-domain data. Chu et al. (2017) proposed two methods for fine-tuning which do not need any modifications to standard NMT architecture. One method is to add domain tags (e.g: '<2domain>') and train the NMT model on the combined corpora from multiple domains. The second method is to fine-tune the model trained on out-of-domain data on the combination of in-domain and out-of-domain data. Britz et al. (2017) proposed three methods for domain adaptation. *'Discriminative Mixing'* method uses a discriminator which is a fully connected layer, to predict the domain tag the current input sentence belongs to. The loss from discriminator and decoder is added and back-propagated which jointly optimizes the network. This makes the encoder encode domain-related features. *'Adversarial Discriminative Mixing'* method is the same as *'Discriminative Mixing'* method except while back-propagating loss, the loss from discriminator is reversed by multiplying it with -1. This makes the encoder encode domain invariant features. *'Target Token Mixing'* does not use a discriminator network but simulates the discriminator by adding domain tags to the target sentence.

Improving the performance of NMT models with additional monolingual data is a common practice especially in low-resource settings. Back-translation (Sennrich et al., 2016) is an effective approach to make use of the target monolingual data. Edunov et al. (2018) conducted various experiments to generate synthetic source sentences from target monolingual data and used it to further train the models. They report that corrupting synthetic source sentences with noise and using that noisy source sentence instead of a clean synthetic source sentence, significantly improve the performance of the NMT models. Multilingual NMT (Johnson et al., 2017) is another popular approach to improve the performance of NMT models for low resource language pairs by augmenting the low resource pairs with high resource language pairs and training a single NMT model.

In this work, we build NMT models with domain adaptation objectives. We experiment with cleaning synthetic in-domain corpus with denoising auto-encoder (Vincent et al., 2008) and Automatic Post-Editing (Pal et al., 2016) objectives. The resulting data is augmented with created in-domain corpus and used to train NMT models. We also experiment with combining multilingual NMT and domain adaptation objectives.

[3] The developed system can be accessible via following link: http://edumt.ngrok.io/

[4] https://amara.org/en/

3 Corpus Creation

We prepare the parallel corpus in educational domain by crawling lecture subtitles. Specifically, we crawl the lecture subtitles from YouTube and lecture transcripts from NPTEL courses[5]. The subtitles crawled from YouTube are of smaller length compared to subtitles crawled from lecture transcripts. We crawl lectures on Programming, Data Structures, Algorithms, Machine Learning, and Artificial Intelligence.

3.1 Data Crawling

- **Crawling Subtitles:** Video lecture subtitles are crawled from NPTEL[6] and MIT OCW[7] YouTube channels. We crawl the data using *youtube-transcript-api*[8] Python package. First, we collect the URLs of lecture videos. Every video has two types of subtitles in English. One is auto-generated by YouTube and the second one is official subtitles uploaded along with the video. We extract only the official subtitles to minimize the amount of noise in the data as much as possible. Table 1 shows the statistics of crawled subtitle corpus.

- **Crawling Lecture Transcripts:** Lecture transcripts are crawled from the NPTEL courses. For a given course, the lecture transcript is made available in PDF format. Every PDF is tagged as 'Verified' and 'To be verified'. We consider the courses whose transcripts are tagged as 'Verified'[9]. We use *pdftotext*[10] Python package to extract text from PDF. After getting the text, we use *sacremoses*[11], a Python implementation of Moses (Koehn et al., 2007) tokenizer to tokenize the data into sentences. Table 2 shows the statistics of crawled transcript corpus.

3.2 Creation of Synthetic Corpus

The crawled data is in English. To prepare the Hindi-Bengali parallel corpus, we use Google translate tool. The lecture subtitles are crawled from YouTube, translated into Hindi and Bengali with the help of YouTube's built-in Google translate tool. The lecture transcripts are translated into Hindi and Bengali with the help of Google translate web interface[12]. Table 3 shows the statistics of prepared synthetic corpus. Table 4 shows language-wise average sentence length of synthetic corpus prepared from the subtitles and transcripts.

Domain	#Videos	#Subtitles
Prog, DS and Algo	838	263,150
ML and AI	678	176,764
Total	1,516	439,914

Table 1: Statistics of corpus prepared from YouTube subtitles. Here, Prog: Programming, DS: Data Structures, Algo: Algorithms, ML: Machine Learning, AI: Artificial Intelligence. #Videos: No. of videos and #Subtitles: No. of subtitles.

Domain	#PDFs	#Subtitles
Prog, DS and Algo	324	46,009
ML and AI	775	109,179
Total	1,099	155,188

Table 2: Statistics of corpus prepared from NPTEL lecture transcripts. Prog: Programming, DS: Data Structures, Algo: Algorithms, ML: Machine Learning, AI: Artificial Intelligence. #PDFs: No. of transcript PDFs and #Subtitles: No. of subtitles.

Domain	#Subtitles
Prog, DS and Algo	309,159
ML and AI	285,943
Total	595,102

Table 3: Statistics of the synthetic corpus. Prog: Programming, DS: Data Structures, Algo: Algorithms, ML: Machine Learning, AI: Artificial Intelligence. #Subtitles: No. of subtitles.

Language	Subtitles	Transcripts
Bengali	11.7	13.96
Hindi	14.6	17.57
English	13.61	16.24

Table 4: Average sentence lengths of synthetic corpora for each language. Subtitles: Data crawled from YouTube lecture subtitles. Transcripts: Data crawled from NPTEL lecture transcripts.

[5] https://nptel.ac.in/course.html

[6] https://www.youtube.com/user/nptelhrd

[7] https://www.youtube.com/user/MIT

[8] https://pypi.org/project/youtube-transcript-api/

[9] 'Verified' transcripts are the transcripts that are post-edited after the automatic transcription is done.

[10] https://github.com/jalan/pdftotext

[11] https://github.com/alvations/sacremoses

[12] translation using Google translate is done between July 2020 to February 2021.

3.3 Creation of Clean Corpus

We create a clean Hindi-Bengali parallel corpus by taking part of synthetic corpus and post-edited by annotators and crowd-sourcing. We remove this data from the synthetic corpus to avoid data duplication when training models. We employ three annotators who are fluent in English, Hindi, and Bengali. We provide English corpus and corresponding Hindi and Bengali translations. The annotators post-edited both Hindi and Bengali data based on the English data. We follow the same method to get data post-edited by crowd-sourcing[13] company also. After a clean corpus is created, we took a random sample of 263 Hindi-Bengali parallel sentences for analysis. We ask 4 people who speak both Hindi and Bengali[14] to score the random sample based on Adequacy and Fluency on a scale of 1-5. For the Hindi part of the sample, the average adequacy and fluency scores are 4.3 and 4.5, respectively. For the Bengali part of the sample, the average adequacy and fluency scores are 4.3 and 4.6, respectively. Based on the manual analysis of the post-edited corpus, we conclude that the post-edited clean corpus is of high quality. Table 5 shows the statistics of the clean corpus.

Domain	**#Subtitles**
Prog, DS and Algo	22,046
ML and AI	18,190
Total	40,236

Table 5: Statistics of the clean corpus. Here, Prog: Programming, DS: Data Structures, Algo: Algorithms, ML: Machine Learning, AI: Artificial Intelligence. #Subtitles: No. of subtitles.

Corpus	**Domain**	**#Sentences**
Synthetic + Clean	Educational	635,338
Samanantar	General	2,501,608

Table 6: Statistics of data used in experiments. Here, Synthetic: Prepared synthetic educational corpus. Clean: Prepared clean educational corpus. Samanantar: Samanantar Hindi-Bengali corpus. #Sentences: No. of sentences.

4 Domain Adaptation

We consider both synthetic and clean Hindi-Bengali educational parallel corpus as in-domain data. Samanantar corpus (Ramesh et al., 2021)[15] is considered as out-of-domain data. Table 6 shows the statistics of data used in experiments. Since there is no standard Hindi-Bengali educational test set is available to test our models, we manually create the test set by translating Hindi part of English-Hindi parallel corpus (AI domain) from Adap-MT shared task (Sharma et al., 2020) into Bengali. We carefully create the test set by avoiding any overlap between the test set and in-domain corpus which is used for training. The prepared test set of size 2,630 sentences is used to evaluate all trained models.

We train two baseline models, namely 'Out-of-domain baseline' and 'In-domain baseline'. The out-of-domain baseline model is trained on Samanantar corpus and the in-domain baseline model is trained on the prepared clean educational parallel corpus. We train two domain adaptation models by following fine-tuning (Chu et al., 2017) method. Specifically, we use the out-of-domain baseline model as the parent model. The parent model is fine-tuned with (i). Clean educational parallel corpus (denoted as 'FT-Clean') (ii). Synthetic + Clean educational parallel corpus (denoted as 'FT-Both'). The reason to build two fine-tuned models is to check whether synthetic corpus is improving model performance or not. Based on the results (ref Table 7) we choose to use both Synthetic and Clean parallel corpus as our in-domain corpus. We also train another fine-tuned model following mixed fine-tuning (Chu et al., 2017) method. Similar to fine-tuned models, the out-of-domain baseline is used as a parent model and fine-tuned with the combination of Samanantar and Synthetic + Clean educational parallel corpus (denoted as 'FT-Both-Mixed').

We also experiment with adding domain tags[16] to source sentence (Chu et al., 2017) (denoted as 'Source Token Mixing') and target sentence (Britz et al., 2017) (denoted as 'Target Token Mixing'). Using these methods, a single model can be trained on both out-of-domain and in-domain data at the same time. This will save time to train the model. In our case, since out-of-domain data size is very large compared to in-domain data, we oversample in-domain data to match the size of the out-of-domain data.

[13] https://xsaras.com/

[14] Please note that there is no overlap between annotators who post-edited the corpus and evaluators.

[15] https://indicnlp.ai4bharat.org/samanantar/#indic-indic

[16] We use ##2GEN, ##2EDU tags to denote general and educational domains respectively.

4.1 Multilingual Domain Adaptation

Multilingual NMT model (Johnson et al., 2017; Sen et al., 2018) is a single model trained for multiple translation directions by combining parallel corpora from multiple languages into a single unified corpus. Multilingual models have shown improvement for language pairs having less corpus. In this work, we experiment with combining domain adaptation objective with multilingual model (Chu and Dabre, 2019) to check whether adding another language to the corpus will improve the model performance or not (denoted as 'FT-Multilingual'). To build this model, we use the Out-of-domain baseline model which is trained on Hindi-Bengali Samanantar corpus, as the parent model. We fine-tune the model on multilingual in-domain corpus obtained by combining Hindi-Bengali, Hindi-English, and English-Bengali corpus[17]. Specifically, we concatenated Hindi-English, English-Bengali, and Hindi-Bengali corpora. Similar to Johnson et al. (2017), we use language tags to denote the target language[18]. Here, the English part of the corpus act as a bridge between Hindi and Bengali.

5 Semi-Synthetic Data Augmentation

Since most of our in-domain data is synthetic, we conduct experiments on automatic corpus cleaning. We experiment with two methods for automatic corpus cleaning, Denoising auto-encoder (Vincent et al., 2008; Edunov et al., 2018) and Automatic Post-Editing (APE) (Pal et al., 2016). We conduct experiments on the Bengali part of the corpus as it is our target language. We use synthetic-clean Bengali sentence pairs from Clean corpus[19] as our training corpus for corpus cleaning experiments. With the APE objective, we train an end-to-end NMT model with synthetic Bengali sentences as input and clean Bengali sentences as the target. Edunov et al. (2018) show that when using back-translated (Sennrich et al., 2016) data to train the NMT model, adding noise to input sentences improve model performance significantly. Similarly, we create a noisy version of source sentences with two types of noise: (i). Randomly dropping word with probability 0.1 (ii). randomly swapping tokens with its neighboring token with probability 0.1 (Edunov et al., 2018). We do not modify the target sentences. We also experiment by combining these two objectives and training a single model which can perform both denoising and automatic post-editing. After training, we use these models to generate clean Bengali sentences from synthetic Bengali sentences. We denote this as 'Semi-Synthetic' corpus since the source (Hindi) is synthetic and the target (Bengali) is automatically cleaned.

The main reason to perform automatic corpus cleaning is to use the resulting clean corpus to improve the performance of the NMT model for the educational domain. To this extent, we repeat the experiment similar to 'FT-Both' which is fine-tuning the model trained on Samanantar corpus with educational corpus. However, now we use Semi-Synthetic corpus along with Synthetic and Clean corpora to fine-tune the model. 'FT-Both + Denoising' denotes the model fine-tuned with clean, synthetic corpora and semi-synthetic corpus obtained from the denoising experiment. 'FT-Both + APE' denotes the model fine-tuned with clean, synthetic corpora and semi-synthetic corpus obtained from the APE experiment. Similarly, 'FT-Both + Denoising + APE' denotes the model fine-tuned with clean, synthetic corpora and semi-synthetic corpus obtained from the experiment combining denoising and APE objectives. The reason to combine the semi-synthetic data with clean and synthetic data is to provide the model with as much data as possible since in-domain data size is less compared to out-of-domain data.

6 Experimental Setup

All the models have trained on the Transformer (Vaswani et al., 2017) architecture. We use 6 layer Encoder-Decoder stacks with 8 attention heads. Embedding and hidden sizes are set to 512, dropout (Srivastava et al., 2014) rate is set to 0.1. The feed-forward layer consists of 2,048 cells. Adam (Kingma and Ba, 2015) optimizer is used for training with 8,000 warm-up steps with an initial learning rate of 2. We use token-wise batching with batch size set to 2048 tokens. For fine-tuned models, the parent model is trained till convergence[20] and the child model is initialized with the last checkpoint from the parent model without resetting any hyper-parameters. All the models are trained

[17] Since we created the educational corpus by translating English to Hindi and Bengali, we have 3-way parallel corpus involving Hindi, Bengali and English languages

[18] We use ##2EN, ##2BN tags to denote English and Bengali respectively

[19] Since we created clean corpus from synthetic corpus, we have synthetic-clean sentence pairs.

[20] Perplexity is used as stopping criterion.

till convergence and checkpoints are created after every 10,000 steps. All the checkpoints are averaged and considered the best parameters for the respective model. We use OpenNMT toolkit (Klein et al., 2017)[21] to train the models. We tokenize the data into subwords with the unigram language model (Kudo, 2018) using SentencePiece (Kudo and Richardson, 2018) implementation. For all the models except 'FT-Multilingual', we learn subword rules on corpus obtained by concatenating in-domain and out-of-domain corpora. The size of subword vocabulary is 50K for both Hindi and Bengali. For the 'FT-Multilingual' model, we learn joint subword vocabulary for Hindi, Bengali, and English by combining all the in-domain corpora and Hindi-Bengali out-of-domain corpora, and the size of joint subword vocabulary is 75K. At the time of decoding, the beam size is set to 5 with no length penalty.

Model	**BLEU**(↑)	**TER**(↓)
Out-of-domain Baseline	17.3	0.608
In-domain Baseline	12.6	0.704
FT-Clean	21.5	0.634
FT-Both	33.6	0.482
FT-Both-Mixed	27.7	0.548
Source Token Mixing	23.0	0.607
Target Token Mixing	18.6	0.692
FT-Multilingual	**34.8**	**0.466**
FT-Both + Denoising	33.5	0.481
FT-Both + APE	33.0	0.493
FT-Both + Denoising + APE	32.7	0.493

Table 7: BLEU and TER scores of all trained models. FT-Multilingual model outperforms all other models with 34.8 BLEU score and 0.466 TER score.

7 Results and Analysis

We test all the models on the prepared Hindi-Bengali test corpus of size 2,630 and report BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) scores, calculated with sacreBLEU (Post, 2018)[22]. Table 7 shows the results of the models[23]. The two baseline models, *viz.* Out-of-domain Baseline and In-domain Baseline performance are the lowest of all the other models. This behavior is expected since there is less relevant data as the models are trained on out-of-domain corpus and small in-domain corpus respectively. However, the models trained with fine-tuning objectives, namely FT-Clean, FT-Both, and FT-Both-Mixed achieve better results than both baseline models. Specifically, FT-Both, the model which fine-tuned with both clean and synthetic in-domain corpus achieved better results than the other two models. Interestingly, FT-Both-Mixed, the model fine-tuned with combining data from in-domain and out-of-domain data, achieves less BLEU score (27.7) than FT-Both (33.6) despite this method showing improvement (Chu et al., 2017) in other cases. In our case, adding the out-of-domain data is not helping the model but when compared to the other two fine-tuned models, it converged faster which suggests that the model is over-fitting. We also observe that adding in-domain data although it is synthetic, is helping the model.

The models, Source Token Mixing and Target Token Mixing performance are less compared to fine-tuned models. Despite a single model jointly trained for both in-domain and out-of-domain and can share information between both the domains, performance on in-domain data is not significant. Both the models outperform baseline models but the Target Token Mixing model achieves less BLEU score (18.6) than the FT-Clean model (21.5). Similar to the FT-Both-Mixed model, adding out-of-domain data is acting as noise which limits the performance of the model on in-domain data.

The model fine-tuned with the multilingual educational corpus (FT-Multilingual) achieve the highest BLEU score of 34.8 and lowest TER score of 0.466 (higher BLEU and lower TER scores are preferable) of all other models. We observe that adding more in-domain data is improving the model performance. In our case, we add English in-domain corpus (i.e. Hindi-English and English-Bengali) to the Hindi-Bengali corpus. Since both Hindi and Bengali synthetic data were prepared from English data, adding English along with Hindi-Bengali helped the model to learn better representations for the Hindi-Bengali pair. This is evident from the experiments with BLEU score of FT-Multilingual model (34.8) improved by +1.2 points than FT-Both model (33.6). Similarly the TER score of the FT-Multilingual model (0.466) improved by -0.016[24] points than FT-Both model (0.482).

[21] https://github.com/OpenNMT/OpenNMT-py/tree/1.2.0

[22] https://github.com/mjpost/sacreBLEU

[23] sacreBLEU signatures:
BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a+version.1.5.1
TER+tok.tercom-nonorm-punct-noasian-uncased+version.1.5.1

[24] Negative sign indicates the improvement as lower TER score is better.

Results from the semi-synthetic in-domain data augmentation models are interesting due to the reason that adding more in-domain data is not improving the performance. This observation is opposite of the observation from the FT-Multilingual model where adding the English part of the parallel corpus is making the model outperform all other models. Although the models, namely FT-Both + Denoising, FT-Both + APE, and FT-Both + Denoising + APE are trained on in-domain corpus twice the size of actual in-domain corpus (since we add the semi-synthetic corpus to clean and synthetic corpus, the size of in-domain corpus become almost doubled) none of the models can outperform FT-Both model (it only trained on clean and synthetic corpus). However, these three models outperform all other models except FT-Both and FT-Multilingual with FT-Both + Denoising model achieving the second-best TER score (0.481). We observe that the Denoising objective is more effective than the APE objective for automatic corpus cleaning. We believe that if more synthetic-clean in-domain sentence pairs are available to train the denoising model, it will improve the quality of the semi-synthetic corpus which, in turn, improves the NMT model.

We conduct a human evaluation on the output of our best model, namely FT-Multilingual. We randomly choose 50 sentences from the test set and given to 4 evaluators[25] along with reference and output of the model and asked to evaluate based on Adequacy and Fluency on the scale of 1-5. The average adequacy and fluency scores are 3.5 and 3.85, respectively. Based on the human evaluation, we conclude that the model can translate educational data with good adequacy and fluency.

8 Conclusion

In this paper, we have explored the problem of building an NMT system in the educational domain for the Hindi-Bengali language pair. Since there is no data available in the educational domain, we created the parallel corpus by extracting from lecture subtitles and transcripts and translating them into Hindi and Bengali. We also create a clean parallel corpus by post-editing the parallel corpus via crowd-sourcing as well as with the help of annotators. We trained Neural Machine Translation models with domain adaptation objectives by training models on publicly available Samanantar Hindi-Bengali parallel corpus and fine-tuned with prepared educational data. We explored various methods to fine-tune the models such as mixed fine-tuning, source token mixing, and target token mixing. We experimented with data augmentation methods by automatically cleaning the synthetic in-domain corpus with denoising auto-encoder and automatic post-editing objectives. The resulting data is combined with prepared in-domain corpus and trained models. We also experimented with combining domain adaptation with multilingual NMT by training a model on Samanantar Hindi-Bengali corpus and fine-tuned with multilingual in-domain corpus obtained by combining Hindi-Bengali, Hindi-English, and English-Bengali in-domain corpora. Since there is no standard test corpus is available, we created Hindi-Bengali educational test corpus through manual translation. We observed that the multilingual model outperformed all other models by achieving 34.8 BLEU and 0.466 TER points. We also conducted a human analysis of the multilingual model by taking a sample of 50 random sentences evaluated based on adequacy and fluency metrics by 4 evaluators. The model achieved average adequacy and fluency scores of 3.5 and 3.85, respectively.

9 Acknowledgement

Authors gratefully acknowledge the support from "Speech to Speech Machine Translation", Sponsored by PSA, Government of India.

References

Ahmed Abdelali, Francisco Guzman, Hassan Sajjad, and Stephan Vogel. 2014. The AMARA corpus: Building parallel language resources for the educational domain. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1856–1862, Reykjavik, Iceland. European Language Resources Association (ELRA).

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*.

Maximiliana Behnke, Antonio Valerio Miceli Barone, Rico Sennrich, Vilelmini Sosoni, Thanasis Naskos, Eirini Takoulidou, Maria Stasimioti, Menno van Zaanen, Sheila Castilho, Federico Gaspari, Panayota Georgakopoulou, Valia Kordoni, Markus Egg, and

[25] These evaluators are the same who evaluated the quality of prepared clean in-domain corpus.

Katia Lida Kermanidis. 2018. Improving machine translation of educational content via crowdsourcing. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Denny Britz, Quoc Le, and Reid Pryzant. 2017. Effective domain mixing for neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 118–126, Copenhagen, Denmark. Association for Computational Linguistics.

Chenhui Chu and Raj Dabre. 2019. Multilingual multi-domain adaptation approaches for neural machine translation. *arXiv preprint arXiv:1906.07978*.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of domain adaptation methods for neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–391, Vancouver, Canada. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. 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.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. 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*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Valia Kordoni, Antal van den Bosch, Katia Lida Kermanidis, Vilelmini Sosoni, Kostadin Cholakov, Iris Hendrickx, Matthias Huck, and Andy Way. 2016. Enhancing access to online education: Quality machine translation of MOOC content. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 16–22, Portorož, Slovenia. European Language Resources Association (ELRA).

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Santanu Pal, Sudip Kumar Naskar, Mihaela Vela, and Josef van Genabith. 2016. A neural network based approach to automatic post-editing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 281–286, Berlin, Germany. Association for Computational Linguistics.

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.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. 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. 2021. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *arXiv preprint arXiv:2104.05596*.

Sukanta Sen, Kamal Kumar Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2018. IITP-MT at

WAT2018: Transformer-based multilingual indic-English neural machine translation system. In *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation: 5th Workshop on Asian Translation: 5th Workshop on Asian Translation*, Hong Kong. Association for Computational Linguistics.

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.

Dipti Misra Sharma, Asif Ekbal, Karunesh Arora, Sudip Kumar Naskar, Dipankar Ganguly, Sobha L, Radhika Mamidi, Sunita Arora, Pruthwik Mishra, and Vandan Mujadia, editors. 2020. *Proceedings of the 17th International Conference on Natural Language Processing (ICON): Adap-MT 2020 Shared Task*. NLP Association of India (NLPAI), Patna, India.

Matthew Snover, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Vilelmini Sosoni, Katia Lida Kermanidis, Maria Stasimioti, Thanasis Naskos, Eirini Takoulidou, Menno van Zaanen, Sheila Castilho, Panayota Georgakopoulou, Valia Kordoni, and Markus Egg. 2018. Translation crowdsourcing: Creating a multilingual corpus of online educational content. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103.

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.