NVIDIA NeMo Offline Speech Translation Systems for IWSLT 2022

Oleksii Hrinchuk*, **Vahid Noroozi, Abhinav Khattar, Anton Peganov, Sandeep Subramanian, Somshubra Majumdar, Oleksii Kuchaiev**
NVIDIA, Santa Clara, CA

Abstract

This paper provides an overview of NVIDIA NeMo's speech translation systems for the IWSLT 2022 Offline Speech Translation Task. Our cascade system consists of 1) Conformer RNN-T automatic speech recognition model, 2) punctuation-capitalization model based on pre-trained T5 encoder, 3) ensemble of Transformer neural machine translation models fine-tuned on TED talks. Our end-to-end model has less parameters and consists of Conformer encoder and Transformer decoder. It relies on the cascade system by re-using its pre-trained ASR encoder and training on synthetic translations generated with the ensemble of NMT models. Our En→De cascade and end-to-end systems achieve 29.7 and 26.2 BLEU on the 2020 test set correspondingly, both outperforming the previous year's best of 26 BLEU.

1 Introduction

We participate in the IWSLT 2022 Offline Speech Translation Task (Anastasopoulos et al., 2022) for English→German and English→Chinese. Due to the limited amount of direct speech translation (ST) data, we mostly focused on building a strong cascade pipeline structured as follows:

- ASR model with Conformer (Gulati et al., 2020b) encoder and RNN-T (Graves, 2012) decoder trained with SpecAugment (Park et al., 2019) which transforms input audio into lower-cased text without punctuation.
- Punctuation-capitalization (PC) model with T5 (Raffel et al., 2019) encoder and classification head which transforms normalized ASR output into standard English text, more suitable for NMT model.
- Ensemble of 4 NMT Transformers (Vaswani et al., 2017) trained with back-translation and right-to-left distillation and fine-tuned on TED talks which translates English text into target language.

We also trained end-to-end models capitalizing on the pre-trained ASR encoder and synthetic translations obtained with the ensemble of NMT models. Our best end-to-end model consisting of Conformer encoder and Transformer decoder lags behind the best cascade by 2.7 BLEU on average, however, it might be preferred for some scenarios of limited resources or latency requirements.

Our systems are open-sourced as part of NVIDIA NeMo[1] framework (Kuchaiev et al., 2019).

2 Data

In this section, we describe the datasets used for training (Table 1). For evaluation, we used the development sets of Must-C v2, as well as the test sets from past IWSLT competitions.

ASR For training our ASR model, we used LibriSpeech (Panayotov et al., 2015), Mozilla Common Voice v6.1 (Ardila et al., 2019), TED-LIUM v3 (Hernandez et al., 2018), VoxPopuli v2 (Wang et al., 2021a), all available speech-to-English data from Must-C v2 (Cattoni et al., 2021) En-De/Zh/Ja datasets, ST-TED (Jan et al., 2018), and clean portion of Europarl-ST (Iranzo-Sánchez et al., 2020).

PC For training our punctuation-capitalization (PC) model, we combined 268M sentences from Europarl (Koehn, 2005), RAPID (Rozis and Skadiņš, 2017), TED (Cettolo et al., 2012), news-crawl, news-commentary English corpora used in WMT 2021 (Akhbardeh et al., 2021) and Wikipedia dump from WMT 2020. After that, we split the data into segments of up to 128 words ignoring sentence boundaries and removed all punctuation and capitalization.

*Correspondence to: ohrinchuk@nvidia.com

[1] https://github.com/NVIDIA/NeMo

Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022), pages 225 - 231

Table 1: Statistics of different datasets used for training. Synthetic datasets are marked with `typewriter` font.

Task	Dataset	Size	Time
ASR	LibriSpeech	281K	960
	CommonVoice v6.1	564K	901
	TED-LIUM v3	268K	454
	VoxPopuli v2	182K	523
	MuST-C v2 ASR	410K	728
MT De	WMT'21 bitext	60M	–
	`WMT'21 BT`	250M	–
	`WMT'21 R2L`	60M	–
MT Zh	WMT'21 bitext	42M	–
	OpenSubtitles	11M	–
	ST En→Zh	640K	1K
ST	MuST-C v2	251K	450
	CoVoST v2	290K	430
	ST-TED	172K	273
	Europarl-ST	33K	77
	`ASR synthetic`	1.3M	2.3K

MT For training our NMT models, we used all available bitext from WMT 2021 (Akhbardeh et al., 2021), as well as its right-to-left distillation and back-translated monolingual data (for En→De only), following Subramanian et al. (2021). After training, we fine-tuned our models on bitexts from Must-C v2 dataset.

ST For training our end-to-end ST models, we used Must-C v2, CoVoST v2 (Wang et al., 2020), ST-TED, and clean portion of Europarl-ST. In addition, we translated English transcripts from ASR datasets with unnormalized transcripts (all datasets, except for LibriSpeech and TED-LIUM v3) to obtain more speech-to-German data.

3 System

In this section, we describe the essential components of our cascade and end-to-end submissions.

Segmentation We relied on voice activity detection (VAD) to transform long TED talks from the evaluation datasets into smaller segments. Specifically, we used `WebRTC`[2] toolkit with frame duration, padding duration, and aggressive mode set to 30ms, 150ms, and 3, respectively. Following Inaguma et al. (2021), we then merged multiple short segments into longer chunks until there were no two segments shorter than a threshold $M_{dur} = 12$ms with the time interval between them below a threshold $M_{int} = 50$ms. We also experimented with other hyperparameters in the vicinity of these values but the resulting average BLEU score on IWSLT test datasets from previous years was lower.

ASR We transcoded all audio data to mono-channel 16kHz wav format and normalized all the transcripts by removing capitalization and all punctuation marks except for apostrophe. We also discarded samples shorter than 0.2s and longer than 24s. As a result, our training dataset contained 1.9M audio segments with the total duration of 3800 hours.

We then trained a large version of conformer-transducer (Gulati et al., 2020a) with roughly 120M parameters, which uses RNN-T loss and decoder (Graves, 2012). The prediction network consists of a single layer of LSTM (Hochreiter and Schmidhuber, 1997) and the joint network is an MLP. All the hidden sizes in the decoder were set to 640.

PC Our punctuation-capitalization (PC) model consists of Transformer encoder initialized with pre-trained T5 (Raffel et al., 2019) and two classification heads — one for predicting punctuation and another for predicting capitalization. Capitalization head has two labels which correspond to whether the corresponding token needs to be upper-cased. Punctuation head has four labels for period, comma, question mark, and no punctuation which correspond to whether the corresponding token needs to be followed by a particular punctuation mark.

To do inference on the text of arbitrary length, we split it into segments of equal `segment length` and compute a sliding window (with a step `step`) product of token probabilities. To reduce prediction errors near the segment boundaries, we discard probabilities of `margin` tokens near the segment boundaries except for the left boundary of the first segment and the right boundary of the last segment. Table 2 illustrates how the described procedure works on a given fragment from Wikipedia.

NMT Our En→De text-to-text NMT models were based on NVIDIA NeMo's submission to the last year WMT'21 competition. We discarded all examples where a sentence in either language is

[2] https://github.com/wiseman/py-webrtcvad

Table 2: Capitalization head inference on a text fragment from Wikipedia with the following parameters: `segment length` = 4, `step` = 1, `margin` = 1. Discarded probabilities near the segment boundaries are highlighted in red.

	bantam	sold	it	to	miramax	books
	bantam	sold	it	to		
U	0.9	0.1	0.1	0.2		
O	0.1	0.9	0.9	0.8		
		sold	it	to	miramax	
U		0.5	0.2	0.1	0.8	
O		0.5	0.8	0.9	0.2	
			it	to	miramax	books
U			0.1	0.1	0.8	0.6
O			0.9	0.9	0.2	0.4
	bantam	sold	it	to	miramax	books
U	0.9	0.1	.02	.01	0.8	0.6
O	0.1	0.9	.72	.81	0.2	0.4
	U	O	O	O	U	U
	Bantam	sold	it	to	Miramax	Books

longer than 250 tokens and where the length ratio between source and target exceeds 1.3. We also applied `langid` and `bicleaner` filtering following Subramanian et al. (2021). After such aggressive filtering, we ended up with 60M parallel sentences and 250M monolingual sentences for back-translation. We then trained four 24×6 NMT Transformers using different combinations of bitext, its right-to-left forward translation, and back-translated monolingual data.

Our En→Zh NMT model differs from En→De in that we used jieba tokenization and OpenCC traditional to simplified Chinese normalization, instead of Moses based tokenization and normalization. We used SentencePiece (Kudo and Richardson, 2018) tokenizer with shared vocabulary trained on a combination of English, Chinese and Japanese. We also did not do ensembling.

After training with news-only data, we additionally fine-tuned all our models on MuST-C v2 dataset which resulted in nearly 4 BLEU score boost on IWSLT test sets for En→De. The ensemble of four such models was used to generate synthetic translations for end-to-end ST model training.

To better adapt our cascade NMT models to possible punctuation-capitalization model artifacts, we altered the source side of fine-tuning dataset by normalizing it and running through the PC model.

End-to-end Our end-to-end model is Conformer encoder followed by Transformer decoder trained on pairs of English audio and German translation. After discarding all segments longer than 24s, we ended up with 740K segments with the total duration of 1180 hours. Adding synthetic translations of ASR datasets with unnormalized transcripts resulted in 2.06M segments with the total duration of 3450 hours.

4 Experiments

4.1 Setup

ASR We trained our Conformer-transducer ASR models for 300 epochs with the same architecture introduced in (Gulati et al., 2020a) for large model with AdamW (Loshchilov and Hutter, 2017) optimizer and Inverse Square Root Annealing (Vaswani et al., 2017) with 10K warmup steps and a maximum learning rate of 2×10^{-3}. Weight decay of 0.001 on all parameters was used for regularization. The effective batch size was set to 2K, and we could fit larger batch sizes via batch splitting for the RNN-T loss.

Time-Adaptive SpecAugment (Park et al., 2020) with 2 freq masks ($F = 27$) and 10 time masks ($T = 5\%$) is used as the augmentation scheme. We also used dropout of 0.1 for both the attention scores and intermediate activations. All predictions were made with greedy decoding and no external language model.

For the tokenizer, we trained and used an unigram SenetencePiece (Kudo and Richardson, 2018) with the vocabulary size of 1024. After training, we averaged the best 10 checkpoints based on the validation WER which led to a small boost in both the ASR (Table 3) and the resulting BLEU scores of the complete cascade (Table 4).

Table 3: Word error rate (WER) of the ASR model evaluated on different test datasets. Values in brackets correspond to evaluation on modified references with all numbers converted into their spoken form.

	Librispeech `test-other`	**MuST-C v2** `tst-COMMON`	
Conf RNN-T	4.81	4.35	(2.51)
+ ckpt avg	4.65	4.21	(2.37)

Table 4: En→De BLEU scores calculated on IWSLT test sets from different years by using automatic re-segmentation of the hypothesis based on the reference translation by `mwerSegmenter` implemented in SLTev (Ansari et al., 2021). Avg Δ computes the improvement over the cascade baseline averaged over 7 test sets.

	2010	2013	2014	2015	2018	2019	2020	AvgΔ
Cascade systems								
Conf RNN-T + punct-capit + NMT	20.0	25.2	21.3	22.5	23.8	22.7	25.1	0
+ ASR checkpoint averaging	21.2	26.0	21.4	23.5	24.5	23.3	25.6	+0.7
+ NMT in-domain fine-tuning	24.5	31.3	26.1	27.6	27.6	26.4	28.8	+4.5
+ NMT repunctuated source	26.0	31.5	26.6	28.2	27.5	27.0	29.7	+5.1
+ NMT x4 ensembling	26.6	32.2	26.8	28.3	28.1	27.3	29.7	+5.5
End-to-end systems								
Conformer enc + Transformer dec	17.6	23.5	19.5	17.8	19.4	16.0	16.9	−4.3
+ ASR encoder init	19.8	25.5	21.6	22.4	22.4	20.4	21.7	−1.0
+ ASR synthetic data	24.5	30.0	25.2	25.3	24.9	24.1	26.2	+2.8
Text-to-text								
WMT'21 NMT model	33.3	35.6	31.7	33.5	31.0	28.6	32.4	+9.4
+ in-domain fine-tuning	35.7	41.2	36.2	38.1	34.7	31.7	35.0	+13.1

PC We trained our PC model for up to 400K updates using Adam optimizer (Kingma and Ba, 2014) and Inverse Square Root Annealing (Vaswani et al., 2017) with 12K warm-up steps and a maximum learning rate of 6×10^{-5}. Dropout of 0.1 was used for regularization.

Despite significant imbalance between no punctuation / capitalization and other classes, we trained with cross-entropy loss which showed to perform well in prior work (Courtland et al., 2020). We then computed F1 scores for both classification heads on IWSLT tst2019 dataset. Our high mean punctuation F1 score of 84.6 and capitalization F1 score of 92.6 suggest that the model does not suffer from the class imbalance inherent in the training data.

NMT We trained our NMT models (Transformer, 24×6 layers, $d_{\text{model}} = 1024$, $d_{\text{inner}} = 4096$, $n_{\text{heads}} = 16$) with Adam optimizer (Kingma and Ba, 2014) and Inverse Square Root Annealing (Vaswani et al., 2017) with 30K warmup steps and a maximum learning rate of 4×10^{-4}. The models were trained for a maximum for 450K steps with a dropout of 0.1 on intermediate activations and label smoothing with $\alpha = 0.1$.

After training, we finetuned all our base NMT models on MuST-C v2 for 3–4 epochs with an initial learning rate of 2×10^{-5}, linear annealing and no warmup.

End-to-end Our end-to-end models (17-layer Conformer encoder, 6-layer Transformer decoder, both with $d_{\text{model}} = 512$, $d_{\text{inner}} = 2048$, $n_{\text{heads}} = 8$) were trained for 50 epochs if starting from random initialization and for 30 epochs if using the pre-trained ASR encoder. Our vocabulary consists of 16384 YouTokenToMe[3] byte-pair-encodings trained on German transcripts of ST corpus.

4.2 Results

English-German Table 4 shows the performance of our baseline En→De system and its modifications on 7 different IWSLT test sets over the years. While all proposed modifications lead to clear improvements in BLEU scores, in-domain fine-tuning of NMT model contributes the most, adding almost 4 BLEU to both cascade and text-to-text.

End-to-end model trained on ST data lags behind the baseline cascade. Utilizing the pre-trained ASR encoder and additional synthetic translation data results in a significant boost of 7 BLEU score, however, the gap between end-to-end and best cascade is still 2.7 BLEU.

The difference of 7.6 BLEU between our best cascade and text-to-text translation of the ground truth transcripts suggests that there is still plenty of room for improvement on both ASR and PC parts of the cascade.

[3] https://github.com/VKCOM/YouTokenToMe

English-Chinese We evaluated our En→Zh submission on the development set of the MuST-C v2 dataset released by the competition organizers. Our cascade which differs by the NMT block only from the En→De cascade achieved 25.3 BLEU which improved to 26.7 BLEU after fine-tuning on re-punctuated in-domain data.

4.3 Discarded alternatives

When designing our submission, we explored a number of alternatives. They did not lead to clear improvement in preliminary experiments and, thus, were not included into the final submission.

ASR For our speech recognition part, we experimented with:

- other models, specifically, CitriNet (Majumdar et al., 2021) and Conformer-CTC;
- training on a subset of data (approximately 2.5K hours) with unnormalized transcripts to remove the necessity of using PC model;
- increasing model size by the factor of 1.5 for each parameter tensor.

Interestingly, using fully convolutional CitriNet model allowed us to transcribe the complete TED talks without need for audio segmentation. Unfortunately, the WER of this model was significantly higher than WER of more powerful Conformer-RNNT which resulted in worse overall performance.

PC For our punctuation-capitalization restoration part, we experimented with:

- training the described above PC model from scratch;
- initializing our encoder with BERT large (Devlin et al., 2019) and MBART50 (Liu et al., 2020) weights;
- replacing classification head with autoregressive seq-to-seq model following Cho et al. (2017).

NMT We experimented with more elaborate decoding mechanisms such as shallow fusion with external language model and noisy channel reranking (Yee et al., 2019) but got similar results at the cost of significant computation overhead. Note that both De language model and backward De→En model were not fine-tuned on in-domain data unlike the forward En→De model.

5 Conclusion

We present NVIDIA NeMo group's offline speech translation systems for En→De and En→Zh IWSLT 2022 Tasks.

Our *primary* cascade system consists of Conformer RNN-T ASR model, followed by Transformer-based PC and NMT models. To improve over the baseline, we utilize checkpoint averaging, in-domain fine-tuning, adaptation to PC artifacts, and ensembling. The resulting submission outperforms the last year's best (Wang et al., 2021b) by 3.7 BLEU on IWSLT 2020 test dataset. However, it is worth noting that this year more data was available for training.

Our *contrastive* end-to-end model consists of Conformer encoder and Transformer decoder and translates speech directly into the text in target language. The performance of such model trained on available ST data was almost 10 BLEU worse comparing to cascade. We managed to shrink this gap to 2.7 BLEU by capitalizing on strong ASR and NMT components of our cascade via pre-training and synthetic data generation. Due of its size and simplicity this model may be preferred for some scenarios, such as simultaneous speech translation.

Acknowledgments

The authors would like to thank Boris Ginsburg for many useful discussions over the course of this project and anonymous reviewers for their valuable feedback.

References

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R Costa-jussà, Cristina España-Bonet, Angela Fan, Christian Federmann, et al. 2021. Findings of the 2021 conference on machine translation (wmt21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88.

Antonios Anastasopoulos, Luisa Bentivogli, Marcely Z. Boito, Ondřej Bojar, Roldano Cattoni, Anna Currey, Georgiana Dinu, Kevin Duh, Maha Elbayad, Marcello Federico, Christian Federmann, Hongyu Gong, Roman Grundkiewicz, Barry Haddow, Benjamin Hsu, Dávid Javorský, Věra Kloudová, Surafel M. Lakew, Xutai Ma, Prashant Mathur, Paul McNamee, Kenton Murray, Maria Nădejde, Satoshi Nakamura, Matteo Negri, Jan Niehues, Xing Niu, Juan Pino, Elizabeth Salesky, Jiatong Shi, Sebastian Stüker, Katsuhito Sudoh, Marco Turchi, Yogesh Virkar, Alex Waibel, Changhan Wang, and Shinji Watanabe. 2022.

FINDINGS OF THE IWSLT 2022 EVALUATION CAMPAIGN. In *Proceedings of the 19th International Conference on Spoken Language Translation (IWSLT 2022)*, Dublin, Ireland. Association for Computational Linguistics.

Ebrahim Ansari, Ondřej Bojar, Barry Haddow, and Mohammad Mahmoudi. 2021. SLTEV: Comprehensive evaluation of spoken language translation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 71–79, Online. Association for Computational Linguistics.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M Tyers, and Gregor Weber. 2019. Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Roldano Cattoni, Mattia Antonino Di Gangi, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. Must-c: A multilingual corpus for end-to-end speech translation. *Computer Speech & Language*, 66:101155.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit3: Web inventory of transcribed and translated talks. In *Conference of european association for machine translation*, pages 261–268.

Eunah Cho, Jan Niehues, and Alex Waibel. 2017. Nmt-based segmentation and punctuation insertion for real-time spoken language translation. In *Interspeech*, pages 2645–2649.

Maury Courtland, Adam Faulkner, and Gayle McElvain. 2020. Efficient automatic punctuation restoration using bidirectional transformers with robust inference. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 272–279.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020a. Conformer: Convolution-augmented Transformer for speech recognition. In *Proceedings of Interspeech*, pages 5036–5040.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. 2020b. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Esteve. 2018. Tedlium 3: twice as much data and corpus repartition for experiments on speaker adaptation. In *International conference on speech and computer*, pages 198–208. Springer.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hirofumi Inaguma, Brian Yan, Siddharth Dalmia, Pengcheng Guo, Jiatong Shi, Kevin Duh, and Shinji Watanabe. 2021. Espnet-st iwslt 2021 offline speech translation system. *arXiv preprint arXiv:2107.00636*.

Javier Iranzo-Sánchez, Joan Albert Silvestre-Cerda, Javier Jorge, Nahuel Roselló, Adria Giménez, Albert Sanchis, Jorge Civera, and Alfons Juan. 2020. Europarl-st: A multilingual corpus for speech translation of parliamentary debates. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8229–8233. IEEE.

Niehues Jan, Roldano Cattoni, Stüker Sebastian, Mauro Cettolo, Marco Turchi, and Marcello Federico. 2018. The iwslt 2018 evaluation campaign. In *Proceedings of IWSLT*, pages 2–6.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of machine translation summit x: papers*, pages 79–86.

Oleksii Kuchaiev, Jason Li, Huyen Nguyen, Oleksii Hrinchuk, Ryan Leary, Boris Ginsburg, Samuel Kriman, Stanislav Beliaev, Vitaly Lavrukhin, Jack Cook, et al. 2019. Nemo: a toolkit for building ai applications using neural modules. *arXiv preprint arXiv:1909.09577*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Somshubra Majumdar, Jagadeesh Balam, Oleksii Hrinchuk, Vitaly Lavrukhin, Vahid Noroozi, and Boris Ginsburg. 2021. Citrinet: Closing the gap between non-autoregressive and autoregressive end-to-end models for automatic speech recognition. *arXiv preprint arXiv:2104.01721*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an ASR corpus based on public domain audio books. In *Proceedings of ICASSP*, pages 5206–5210. IEEE.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. 2019. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*.

Daniel S Park, Yu Zhang, Chung-Cheng Chiu, Youzheng Chen, Bo Li, William Chan, Quoc V Le, and Yonghui Wu. 2020. Specaugment on large scale datasets. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6879–6883. IEEE.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Roberts Rozis and Raivis Skadiņš. 2017. Tilde model-multilingual open data for eu languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 263–265.

Sandeep Subramanian, Oleksii Hrinchuk, Virginia Adams, and Oleksii Kuchaiev. 2021. Nvidia nemo neural machine translation systems for english-german and english-russian news and biomedical tasks at wmt21. *arXiv preprint arXiv:2111.08634*.

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 *Proceedings of NeurIPS*, pages 5998–6008.

Changhan Wang, Morgane Riviere, Ann Lee, Anne Wu, Chaitanya Talnikar, Daniel Haziza, Mary Williamson, Juan Pino, and Emmanuel Dupoux. 2021a. Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation. *arXiv preprint arXiv:2101.00390*.

Changhan Wang, Anne Wu, and Juan Pino. 2020. Covost 2 and massively multilingual speech-to-text translation. *arXiv preprint arXiv:2007.10310*.

Minghan Wang, Yuxia Wang, Chang Su, Jiaxin Guo, Yingtao Zhang, Yujia Liu, Min Zhang, Shimin Tao, Xingshan Zeng, Liangyou Li, et al. 2021b. The hw-tsc's offline speech translation systems for iwslt 2021 evaluation. *arXiv preprint arXiv:2108.03845*.

Kyra Yee, Nathan Ng, Yann N Dauphin, and Michael Auli. 2019. Simple and effective noisy channel modeling for neural machine translation. *arXiv preprint arXiv:1908.05731*.